TAS 链白皮书

区块链不可能三角最优解

TAS 链团队

申明

本文仅是 TAS 设计探讨性的白皮书，描述了 TAS 平台的现有技术以及未来开发目标，用于技术社区讨论商议。截至目前，TAS 平台仍在开发中，其设计理念、共识机制、加密算法、开发代码和其它技术细节均可能会不断更新和更改。本文内容可以认为是截止书写时刻的 TAS 平台有关的最新信息，但它未必就是完整，完备的。可以预见，随着 TAS 平台内测，公测，以及更多的 DAPP 上链，项目组会根据需求情况不断调整和更新现有技术方案。

TAS 平台将会创造一个开放，合作，创新的技术社区生态，我们欢迎各类经济学家、技术专家、法律专家、社群运营者以及全球开发者共同参与 TAS 的技术讨论，共同参与和推动区块链公链技术的落地以及进步。

目录

为什么要做 TAS

2006 年，我们考虑把数字版权和 P2P 网络通过代币激励有效结合起来（受理专利号 CN1889119A，“基于点对点文件交换系统的数字产品销售与分享方法”[16]），版权方可以便捷地把数字产品提交到 P2P 网络，并通过 0.1 模式激励矿工在线，即用户下载一首歌支付 1 虾币，版权方得 0.9，矿工得 0.1。基于这个理念，我们创办了虾米音乐（www.xiami.com），中国 TOP3 的音乐分享平台，高峰时日活跃用户（DAU）700 万，基于用户在线行为的个性化推荐长期雄踞中国第一。

2008 年，中本聪划时代的提出了 “一种点对点的电子现金系统” [1]，并在白皮书公布 3 个月后提交了源代码。这份代码运行至今，无论是 BTC 本身的价值还是 BTC 的底层技术框架-区块链，都对人类社会的发展产生了前所未有的影响。2013 年，Vitalik Buterin 创办的以太坊[2]把可编程性引入到区块链，在大幅提高了全球金融流通性的同时，让越来越多的人参与到区块链大生态共建中。

区块链发展到今天，有足够的时间让我们从 P2P 网络、去中心化、共识机制、代币经济、生态建设等各个维度做深入的思考并作出探索。相比缩短了时间和空间距离的互联网，以解决异构网络信任问题为首要任务的区块链技术有着更为广阔的应用场景。无论是金融、数字版权、电商、中介、电子票据，采用区块链技术都能通过减少中间环节大幅节省成本和提高公信力。如果说一千亿美金市值的比特币是数字货币的基石，五千亿美金的数字货币总市值代表了社会对区块链技术发展的期许，则商用 DAPP 一旦爆发，有理由相信这是一个超过五万亿美金的超级市场。构建大规模商用 DAPP 不但需要去中心化和高安全的共识机制，同时也需要这种机制是高性能和低能耗的。另外，也需要逻辑安全、低成本和用户友

好的高级智能合约系统，以支持各个行业复杂的商业规则和业务逻辑。

观察目前在运行中和仍处在研发阶段的公链，我们发现基于 POW 的比特币和以太坊很好的解决了去中心化和安全性问题，但是由单纯 POW 带来的高能耗[6]和低性能几乎代表了硬币的另一面。我们也观察了一些基于拜占庭容错框架的公链，通过深入的分析，我们认为单纯拜占庭容错更多是解决分布式系统的一致性问题，而对区块链要解决的去中心化场景下的信任问题，并不是最合理的解法。基于拜占庭容错的公链，一般只能有 50 个节点可以参与铸块（实际可能更低），不然节点间的协同通讯复杂度就会呈指数级上升而导致无法在较短的时间窗口内做出最优解。基于 DPOS 的公链有类似的问题，议会制的铸块方式和比特币人人可自由参与的基本理念相去甚远，同时还会带来物理 DDOS 攻击之类的问题。

回溯到比特币的源头，我们认为一条良好的公链是以如下假设为前提的：

- **任何节点可以便捷的加入到网络和离开；**
- **任何节点都有铸块的权力；**
- **对于一个较大数量的群体来说，在良好的激励体系下，好人总是多于坏人；**
- **坏人作恶的成本应大于他的收益。**

另外，如果一条公链的最终目的是构建大规模的商业化应用，则它同时也应该是**高性能**和**低能耗**的，以保证商业应用的可持续发展。从这些角度看，我们发现目前国内外的公链都难以达到要求。基于此，我们设计了 TAS chain。

什么是 TAS

TAS 是去中心化、高安全、低能耗和可编程的区块链公链。比特币的 POW

机制已被时间和价值证明其数学上的不可破解性，在保留 POW 的同时，我们引入 POS 里最具技术领先性的 VRF 来解决去中心化状态下的低性能和高能耗问题。对 DSS（去中心化、安全、可扩展性）不可能三角模型，我们认为去中心化大于安全，而安全又大于性能（可扩展性），即只有在保证了高优先级特征的前提下，去提升低优先级能力才有意义，而不是放弃或削弱高优先级的特征去单纯提升低优先级能力。举个例子，如果削弱去中心化和安全去提升性能，本质上仍是在解决几个云计算中心之间的一致性问题。而区块链技术最激动人心的划时代突破，是解决异构节点之间的信任问题而非任何其它。

相比人类社会里黄金的挖矿，构建于一般等价物之上的商业活动是更精细化和复杂的模型。同样的道理，TAS 的白皮书不仅仅是数学证明和工程技术架构，而是对孵化商用 DAPP 落地的一整套解决方案。我们试图构建一个人人可参与的高安全去中心化网络，基于这个网络可以用很低的成本开发可持续发展的商业应用。同时通过一系列数学、密码学和工程技术的引入，让这个网络在保证去中心化和高安全的同时，也是高性能和低能耗的。

我们认为，如果一个区块链系统能够在保证去中心化和高安全的前提下，性能达到 1000TPS+且整个系统是低能耗的，就可以快速推进大型商用 DAPP 的落地和爆发。在此，我们给出 TAS 的设计目标：

- **去中心化：每个节点能很便捷的参与铸块和离开，整个系统具有良好的抗物理 DDOS 能力。**
- **高安全 :如果一个系统对双花攻击、长程攻击（私自挖矿）、无利害关系、女巫攻击和 51%攻击都有数学可证明或经济博弈的解决方案，则我们认为这个系统是高安全的。因为所有的攻击最终都会汇总到这几种方式永**

久或临时的伤害系统和个体。针对最严重的 51%攻击，我们引入系统健康度概念，交易用户通过简单观察即可发现系统是否处于异常状态，并通过暂停交易或延迟交易的最终确认块保证交易的安全性。

- **高性能和快速确认：TPS 达到 3000 以上。平均 3 秒出块，10 秒最终确认交易。我们认为这样的指标可以满足 95%的商业应用需求。**
- **低能耗：做一个直观的假设，我们认为平均 1000TPS 的铸块消耗电量在 1 千瓦以下，则这个系统是低能耗的。**

大家知道，区块链有个不可能三角定律，即去中心化，安全性和性能无法三者兼得。**SPOW 共识机制用 VRF 解决去中心化问题，用分组 POW 达到甚至超过 POW 的安全性，同时通过组内协作达到了 POS 的性能。**另外，基于我们多年在大型分布式系统的经验，在提案、验证和出块三个环节都保证了无单点设计，进一步提高系统的性能和鲁棒性。**通过严格的数学论证和工程分析，我们认为 SPOW 共识机制给出了迄今为止全球范围内不可能三角定律的最优解。**

SPOW 共识机制的另一个好处，是更符合真实的异构网络拓扑。跟人类社会有穷人和富人一样，我们认为最好的平等并不是均贫富，而是设定一套穷人富人都可以参与的游戏规则，并根据投入的不同分享不同的收益。回顾传统的 POW，也许中本聪最初的设计里并没有矿机和矿池，而是人人可参与的朴素理念，可随着比特币价格的日益上涨，POW 的挖矿越来越成为普通节点难以参与的军备竞赛。POS 存在同样的问题，大部分的 POS 直接抛弃了算力，只强调在线率，极端状态下手机和高性能 PC 在 POS 世界里拥有一样的地位。抛弃算力的同时必然引入无利害关系，所以大部分 POS 系统都需要较高的代币质押，有些甚至是以百分比的方式，极大的制约了人人可参与的区块链核心思想。SPOW 承认存在不

同的节点，并把它们分成重节点和轻节点，在铸块过程中两者协同工作，相互促进和监督，并分享收益的不同部分。通过组合 POS 和 POW，SPOW 极大的降低了代币的质押份额，并且设计了一条由设备指纹代替代币质押（设备即押金）的可持续发展路线，进一步降低代币的质押。综合来看，SPOW 结合了 POS 和 POW 两者的优点，并通过组间公平竞争，组内协作出块的共识方式大幅提高了系统的安全性、鲁棒性和性能，为上层智能合约提供了一个稳定可靠的运行环境。详见“SPOW 共识机制”部分。

SPOW 共识机制兼顾广度和深度，一个有海量轻节点参与的广度网络是好人多于坏人的前提。传统的 POW 共识更强调军备竞赛而不是善用现有或闲置设备，而 SPOW 的设计理念则是希望借助海量低廉的设备来保障整个节点网络的安全性和健壮性，基于这个理念我们在 P2P 网络上施行了一些新的技术，包括在 NAT 穿透上相比 RFC3489 提升 50%以上的高连通网络，和高效的组内通讯支持。这些技术为 SPOW 共识机制提供了网络层的物理保证。详见“P2P 网络”部分。

SPOW 共识机制的 TPS 设计目标在 3000 左右，这个量级已经足够处理大多数的中小型商业化应用，但是应对超大规模 DAPP 仍有不足，TAS 借鉴谷歌 MapReduce 和阿里云批量计算思想，设计了分片并行计算框架。考虑到带宽、IO、信任握手和数据合并复杂度等瓶颈，上百万甚至可无限扩展的 TPS 更像是一个概念而不是一个能落地实施的指标。TAS 计划通过分片并行计算框架进一步把 TPS 提升到 20000，我们认为这个量级已可支撑目前人类社会 99%的商业化应用。详见“分片并行计算框架”部分。

我们希望汇聚核心开发团队和社区、生态的力量，对 TAS 的共识模型做严谨的数学验证，并逐步搭建完善整体的技术架构和源码实现，我们的愿景并不只是

实现一条公链，而是希望协同社会和社区，在数字版权、供应链金融、电子票据、众筹、预售、游戏、中介等商用领域合作和孵化 DAPP 在公链的落地，以技术推动区块链生态的发展。

TAS 关键词

不可能三角最优解，分布式协作，VRF，设备即押金

TAS 架构

图 1. TAS 总览图

网络层

采用无连接的 RUDP 代替 TCP 提升性能。

NAT 高穿透率技术。

针对组协作的双层 KAD 网络。

核心层

密码学部分-包括双线性对椭圆曲线、动态沙米尔秘密共享和 VRF 随机数生成。

组内沙米尔秘密共享验证机制。

组内预算力协作机制。

设备指纹技术。

分片并行计算框架。

开放架构的 TVM。

功能层

软分叉算法。

SPOW 共识机制。

动态组成员扩容机制。

设备即押金机制。

智能合约基础组件库。

铸块激励机制。

动态分片计算。

应用层

手机挖矿支持。

自组织治理。

真随机数 API。

系统实时健康度指标。

跨合约协作支持，合约软升级支持。

组创建、组创始化、组铸块和组销毁的生命周期管理。

高性能两层（共识层+并行计算层）TPS 支持。

核心技术

SPOW 共识机制

从比特币诞生至今，学术界和工业界对区块链的共识机制持续研究，诞生了多种机制、协议与算法，但是这些成果都无法破解一个难题：如何同时满足去中心化、安全、性能的共识三角。POW 机制性能低并且高能耗；POS 机制无法兼顾去中心化与性能，质押制度更是最根本的短板；HASH 图与 DAG 更多是解决一致性问题而非准确性问题。这些共识机制一般会考虑三角的部分侧重，但并没有从全局考虑最优解。同时，很多的共识机制太偏单纯的学术化，在互联网工业界大放异彩的分布式系统强调高效协作，这些成熟的技术并没有被很好的引入区块链。另外，很少有共识机制对无利害关系、51%攻击、女巫攻击、长程攻击等各种安全问题做全面的分析和论证。

在 TAS 中，我们设计了一种全新的共识机制 SPOW，它结合了 VRF+POW，并引入了分布式系统的分片、高并发协作、预处理等技术、具有如下优点：

在去中心化方面，它比比特币更佳，任何可联网设备均可成为节点。并设计了轻节点和重节点机制兼顾广度和深度。

在安全性方面，天然抗无利害关系，同时需要控制 95%的节点才能发动类似比特币的 51%算力攻击。

在性能方面，在不做分片等优化的情况下，它与 EOS 基本相当。

SPOW 包含一系列数学理论工具、密码学算法及完备的协议，但其核心机制可以精简成如下过程：

矿工节点被分组，通常 100 人为一组。

每次出块时，由 VRF 机制选择出块组，保证了出块组的不可预测、不可选择和不可隐藏。

组内通过 POW 算力协作出块，选出前 K 个满足算力条件的节点并行出块，通过算力协作和并行高速出块提高了系统的性能和鲁棒性。同时通过巧妙的协议设计，我们使得 POW 选举出块节点的行为，在本组空闲时间即已完成，在真正出块时，只需在组内达成轻量级验证共识，以多通道并行流水线方式快速出块。

通过与前沿公链的对比，我们可以看出 SPOW 的全面优势：

与 BTC 相比，SPOW 在去中心化与安全性上更强，且性能大幅提升，能耗大幅降低。

与 EOS 相比，SPOW 在性能上大致相当，但在去中心化与安全性上有明显的优势。我们认为，区块链系统的去中心化>安全性>性能。

与 DFINITY 相比，SPOW 在去中心化上与之相当，但是 SPOW 的组内被设计成一个小的分布式协作体，组员之间通过分工协作极大的提高了性能和鲁棒性。组内 POW 机制不但提升了安全性，且和设备指纹技术的结合基本解决了 POS 的质押问题。我们认为比特币那种不需要质押，人人可自由参与和退出的机制更代表区块链的去中心化精神。

与 DCR 相比，同为 POW+POS 的方案，SPOW 在性能上有明显的优势。

通过与主流及新兴公链的比对及数学与形式化的论证，可以得出结论，SPOW 共识机制是迄今为止 “不可能共识三角” 问题的最优解，领先于目前所有已知

的共识机制。

SPOW 机制的完整细节如下：

基本概念与符号

p　　大素数，如 256bit 的素数。密码学参数。

$GF(p)$　　基于p的有限素域

节点。我们把用户设备上运行的一个客户端进程称为节点。根据设备的算力情况，用户可以将节点的属性设置为轻节点和重节点两种：

- 重节点： 计算型 PC，矿机等
- 轻节点： 普通 PC，手机，机顶盒，移动设备，嵌入式设备等

矿工。普通用户通过注册，可以加入参与分布式记账。设 M 为所有矿工的集合，把它们分别添加标记1，2，$\cdots \in |\mathrm{M}|$。根据职能分类为：

- 提案矿工（proposer）：参与 POW 算力计算，负责给出区块提案（proposal）。
- 验证矿工（verifier）： 以组协作方式工作，对提案矿工所给出的候选区块做有效性验证，并在组内达成出块共识。

组。在任何给定时间，一些或所有$\mathrm{i} \in \mathrm{M}$被排列成一个或多个子集$\mathrm{G}_1, \mathrm{G}_2, \cdots \in M$，称为组。我们将提案矿工和验证矿工按系统预设的比例组成一个工作组。每个工作组具有相同的组规模$\mathrm{n} = |\mathrm{G}_i|$。

槽(slot)。区块铸块时间。Slot 轮数与区块高度对应：第r轮 slot 铸出块高(height)为r的区块。若第r轮 slot 铸块失败，则块高为r的区块不存在（$r+1$区块 prehash 指向$r-1$区块)。

纪元（epoch)。一个纪元包含若干个槽（slot)，由相应的系统参数设定。

父亲组。当申请成为矿工的候选者数量上满足建组条件后，需要由一个工作组发起组建新组操作，该工作组称为新组的父亲组。父亲组可以为新组指定一些特殊属性，比如新组的生效纪元等等。

组存续周期。是指工作组的生效纪元至失效纪元之间的时间。由系统参数设定。当时间到达工作组失效纪元后，工作组将被解散。

矿工健康指数。这个指数是多因素的函数，目前我们暂时考虑以下几个因素（权重从高到低排列）：

- 参与铸块（提案或验证）个数，
- 参与率(实际参与次数/理论参与次数)，
- 设备健康度（通过设备指纹获得)，
- 曾加入过的组个数（在线时长)，
- 权益证明。

未来可能会考虑更多的因素进行扩展。

新矿工在健康指数很低时，建议使用权益证明提高自己的健康指数。随着对系统

的贡献增加，健康指数高到一定程度可以不用做权益抵押。因为其他因素权重占比更高，到后期权益证明对 SPOW 机制中的随机选取几乎没有影响。

对于系统的正常运行，需要对全体矿工有一定的假设要求：

1. 系统中肯定有诚实矿工和恶意矿工
2. 诚实矿工比恶意矿工多
3. 矿工的理性行为受经济利益的驱使

开放的参与机制

SPOW 共识采用开放型的参与机制。在这机制里，普通用户可以通过申请加入系统成为矿工；矿工也可以选择通过申请注销自己的矿工身份，回归为普通用户。对此，我们设计了特殊的智能合约：矿工申请合约和矿工注销合约，并写入创世块中，供用户调用。

矿工的注册。普通用户可以提交矿工申请合约来申请成为矿工，申请时用户需指明矿工的类型（Type：提案矿工或验证矿工）。每个矿工都有自己唯一的身份号

$$\mathrm{ID} = \mathrm{trans\_hash}(\mathrm{pk})$$

其中pk是该矿工作为普通用户的公钥, trans_hash 现在采用输出 256bit 的哈希函数。普通用户的公钥是与它的矿工 ID 一一对应的，而且在整个系统内也是唯一的。矿工申请合约将（pk，Type）写入区块链中。该合约针对矿工类型有不同的处理：

提案矿工：不需要权益抵押

验证矿工：健康指数较高，可以免保证金证明，否则，账户中若干权益抵押

作为保证金证明。

原则上，我们希望重节点上的矿工担任提案矿工，轻节点上的矿工担任验证矿工。

矿工的注销。矿工可以提交矿工注销合约来申请退出 SPOW 共识。同样的，矿工注销合约会写入区块链中，若该矿工有保证金证明，判断该矿工成组状态，若未成组，则直接释放保证金给用户；否则等该矿工所在组存续周期过期时，释放保证金给用户。

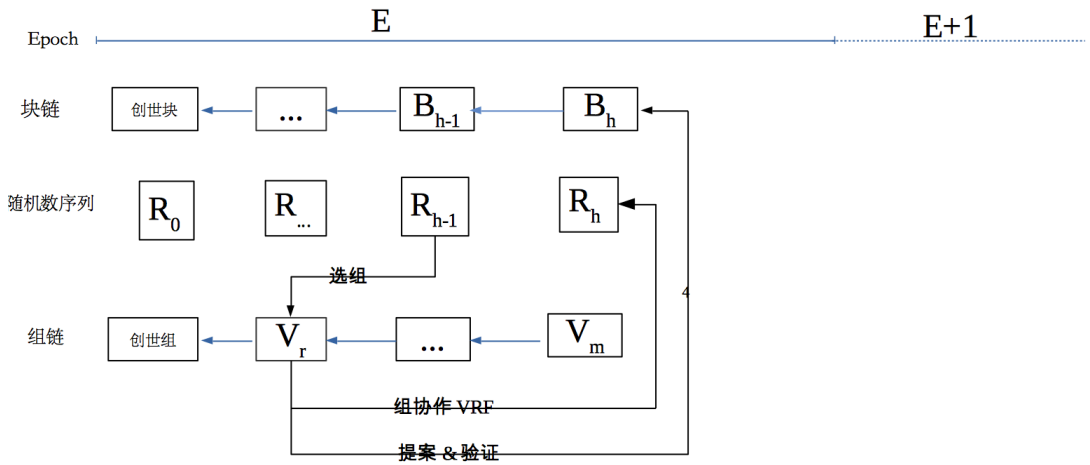

图 2. SPOW 铸块流程图

SPOW 共识对重节点矿工，轻节点矿工随机采样，形成若干个工作组，组协作 VRF 算法产生的真随机数来确定当前 slot 的工作组，组内成员通过算力预算（POW 算法），给出若干个候选区块提案，以组协作方式来完成候选区块验证，并达成组内共识向组外广播。

出于数据的不可篡改性，以及可追溯性，如上图所示，SPOW 机制采用双链模型，区块链和工作组链，来记录整个数据链产生过程。同时考虑组内矿工可能会形成组内携手作恶，所以组有存续周期，到期后会解散重新进入随机采样分组流程。所以当系统进入稳定状态，每当进入一个新纪元（epoch）都可能有新组

生效，老组解散的情况发生，每个纪元的工作组列表会发生相应变动。如下图所示。

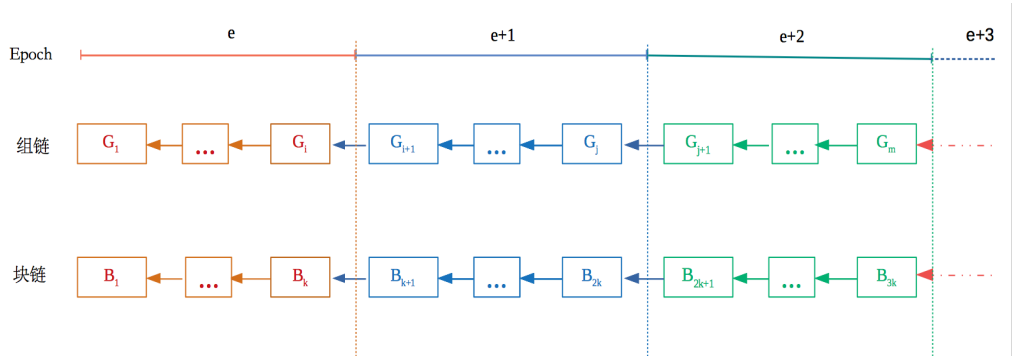

图 3. SPOW 组周期解释图

经济奖励模型

TAS 的设计必须实现对矿工的经济激励，让他们的付出有所收益，以此系统才会良性发展。 经济激励模型必须做到让所有矿工可根据自身付出的劳动获取相应的报酬，可预期，可审计，可追溯，公平公正却又不可篡改。

上述的阐述更像是一个商业的劳动报酬合约的内容，智能合约正是为此而生。自然而然的，智能合约成为实现经济激励模型的最好方法。

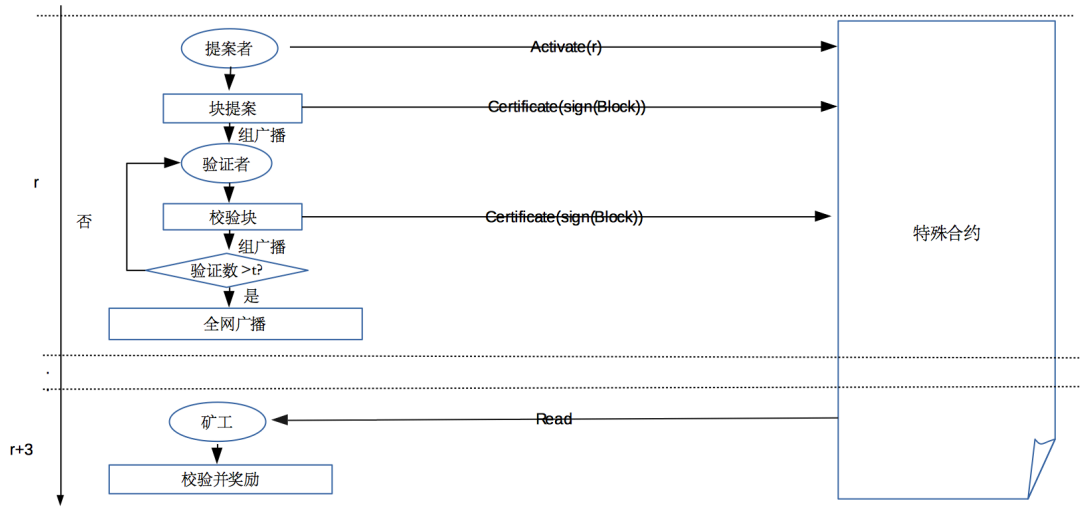

图 4. SPOW 经济奖励流程图

TAS 系统在每一轮铸块开始启动一个特殊的智能合约，参与铸块（包括提案和验证）的每位矿工，在完成自己对本轮的计算工作后，携带相应工作证明的凭证 call 该合约，合约为其记录工作证明，并计算出奖励份额。 在一定时间窗口后，系统自动触发该合约的奖励逻辑，将奖励打入矿工账户。

采用智能合约实现经济激励的设计，不仅可以实现安全公正的经济激励，同时为评估每个矿工节点的健康指数等参数提供了重要的数据基础，根据这些数据统计，我们可以轻易地识别出系统的积极矿工和消极矿工，而矿工的健康指数可以直接对矿工后续继续参与 SPOW 计算有着正向反馈的功效。

组链模型

组块的数据结构包含，但不限于以下信息：

- ✧ Type： 矿工组类型（提案组或者验证组）
- ✧ Hash： 组块哈希值
- ✧ GIS： parent 组（定义见建组检查章节）指定的信息
  - ▫ Activation： 组生效的纪元
  - ▫ Deactivation： 组失效的纪元
  - ▫ MinerArray[]： 候选矿工数组
    - • pubk ： 矿工对应的用户公钥
    - • ID： 矿工的身份号
- ✧ Signature： parent 组对 GIS 的签名。
- ✧ Prehash： parent 组块哈希值

✧ gpk： 组公钥

建组检查

TAS 系统以 CheckInterval（系统参数）为周期来做建组检查。假设当前第r轮 slot，若r 是 CheckInterval 的倍数，则在铸块完成后，当前工作组需要进行建组检查。考虑链同步的问题，我们假设系统网络广播需要 d 个 slot 的时间，那么可以认为前d块之前的区块已经一致。即块高小于等于$r-d$的区块全网已经同步。组成员通过查询区块数据中的矿工申请合约，按一定条件筛选，得到新工作组的矿工候选人数量满足建组要求。则由该工作组作为父亲组，发起新组创建的提案。

新组提案

当前工作组作为父亲组，对新组提案，主要任务是确定新组成员人选。主要策略是根据候选人的健康指数进行排序，以当前区块的真随机数为种子产生伪随机数序列随机挑选，确保选出的新组成员的健康指数分布与所有候选人的健康指数分布类似。即保证有部分高健康指数的组员的同时会携带一些低健康指数的组员。从分组选人上大幅降低“僵尸攻击”的概率。

另外，当前工作组必须为新组指定生效纪元。比如为当前纪元 e 之后的第 2 个纪元

GIS.Activation = e+2

指定新组的失效纪元。由系统预先设定的参数 ValidPeriod 确定。

GIS.Deactivation = GIS.Activation + ValidPeriod

由于上述候选人序列，选择新组成员策略都是可审计的。所以当前工作组内

的诚实矿工，计算出来的组块内容应该都是一样的。所以可以由组内每个矿工分别各自出新组块，对自己的新组块签名，然后发给组内其他成员。组员在收到大于等于门限t个的相同 hash 的新块，可以用它们的组员签名恢复出组私钥对该组块进行签名。作为新组的父亲组，需要调用创世块中的分红智能合约模版，为新组创建分红合约，并公布该合约，这些合约最终会写入区块中。

新组创建

由于组成员是去中心化网络上的对等节点，现实中某些时刻，节点因种种原因不在线不可避免，比如网络信号不好，恶意节点故意不作为等等。所以我们设计验证组需要有(t,n)门限签名的能力，这里n是组成员数，t是恢复的门限值，通常$t \leq n$。即有含门限t个以上的组员对消息认可签名，就能代表全组n个成员对消息的认可，并能恢复出全组对此消息的认可签名。这里我们采用了去中心化的 Shamir 秘密共享算法，来产生各个组员的组内签名私钥S_i，组内签名公钥mpk_i，以及代表组共识的组私钥对应的组公钥gpk，得到上述密钥，并对组公钥gpk达成共识，才算完成组创建。

传统的 Shamir 密钥分享技术[20]，可以看过一个需要“中心”的过程。即它首先要有个“分发者”，“分发者”决定采用哪个秘密多项式，然后将秘密分解，发放给其他人。而区块链网络是去中心化的网络，每个节点都是平等，对等的节点，不应产生中心化的 “分发者”。另外，分发者是先于其他节点知道秘密的，如果分发者在过程中做恶，是很难预防的，因此我们采用了去中心化的密钥分享算法。其核心思想就是：让每个组员都成为“中心”（即秘密分发者），那结果就是没有物理上的“中心”。但会在组的逻辑层面上形成一个初始秘密SK（该秘密每个成员均不知晓）。每位组员通过下述步骤的前三步会形成一个组逻辑层面的分享秘密S_i。类似传统 Shamir 密钥分享算法，分享秘密集合$\{S_i\}$具有门限恢复初始秘密SK的能力，即$\{S_i\}$中任意不少于t个分享秘密均能恢复组初始秘密SK，任意少于t个分享秘密均无法得到组初始秘密SK的任何信息。

具体执行步骤如下：

1. 每个组员自行选取各自的秘密多项式$f_i(x) \in GF(p)[x]$,
$$f_i(x) = a_{i,0} + a_{i,1}x + a_{i,2}x^2 + \cdots + a_{i,t-1}x^{t-1}$$
其中多项式系数$a_{i,0}$，$a_{i,1}$，$a_{i,2}$，$\cdots, a_{i,t-1} \in GF(p)$, 均为每个组员自取的随机数。则每人的初始秘密$sk_i = f_i(0) = a_{i,0}$，以$sk_i$作为私钥，计算对应的公钥$pk_i$。
2. 每个组员计算给其他组员的分享秘密并将该分享秘密发给对应的组员。即：第 i 个组员，计算$S_{i,j} = f_i(ID_j)$，发给第 j 个组员。其中$i = 0,1,2\cdots,n;\ j = 0,1,2\cdots,n$;同时把自己的公钥$pk_i$也发给其他组员。
3. 当组员收集齐其他组员发给自己的分享秘密后, 计算所有收到的分享秘密之和$S_i = \sum_{j=1}^{n} S_{i,j} = \sum_{j=1}^{n} f_j(ID_i)$，计算$gpk = \sum_{i=1}^{n} pk_i$。
各自将自己计算所得的组公钥gpk对全网广播。
4. 每个组员计算组内签名私钥S_i所对应的公钥mpk_i，并将组内签名公钥mpk_i告知给组内其他组员。

注意，第 2 步的组间成员间通讯，需要考虑加密通讯，防止被监听。在新组块信息 MinerArray 里，记录着组内所有组员的普通用户公钥 pubk，我们以此作为 ECDH 密钥交换做加密通讯。

经过上述步骤，每个组员获得了组内签名私钥S_i，组内签名公钥mpk_i，以及组私钥SK对应的组公钥gpk。而组逻辑层面还隐含着获得的组私钥$SK = \sum_{i=1}^{n} sk_i$，因为每个组员只知道自己的初始秘密sk_i，故没法知道SK的值。

下面对组员的组内签名私钥$\{S_i\}$具有门限恢复组私钥SK进行证明：

正确性证明：

对任意$k \geq t$,由于组员顺序不影响结果，不妨假设前k个组员。

令$F(x) = \sum_{i=1}^{n} f_i(x)$,通过k个组员组分享秘密的 Lagrange 插值多项式$G(x) = \sum_{i=1}^{k} S_i \prod_{j=1, j\neq i}^{k} \frac{x - ID_j}{ID_i - ID_j}$，容易知：

对$1 \leq i \leq k$,均成立

$F(ID_i) = \sum_{j=1}^{n} f_j(ID_i) = \sum_{j=1}^{n} S_{j,i} = S_i$,

$G(ID_i) = S_i$

令$H(x) = F(x) - G(x)$,容易知道$H(x)$是最高次数不超过$k-1$次的多项式，而

$H(ID_i)=0$对$1\le i\le k$均成立。所以$H(x)\equiv 0$。即$F(x)=G(x)$。所以通过k个组员组分享秘密的 Lagrange 插值多项式即是组的秘密多项式。计算$G(0)=F(0)=\sum_{i=1}^{n} f_i(0)=SK$

安全性证明：

由于$F(x)=\sum_{i=1}^{n} f_i(x)$是最高次数为$t-1$次的多项式，多项式系数$t$个，而当$k<t$时，只能构建$k$个等式，由线性代数知识可知，方程个数小于未知数个数时，无法求解出未知数（解不唯一）。即无法确定多项式系数，从而无法确定$F(x)$。所以等$k<t$时，无法得到组初始秘密SK的任何信息。

全网节点（包括新组成员节点）在收到相同的新组组公钥，达到门限t个以上时，才认为新组创建成功，组公钥为真。将新组块写到本地的组链上。

组异步创建处理

每次创建新工作组，均需要由父亲组协商决定新组的候选人列表，并通知新组成员，由新组成员自行达成组内密钥创建。而申请加入组和真正执行组创建这两种行为是异步的，由上述知道新工作组创建时，要求所有组员同时开始发送分享秘密，但在实际情况中，这个条件变得很苛刻，因为只要有一个成员不在线，就无法完成建组工作， 因而会大大降低建组成功的概率。为了解决这个问题，SPOW 共识从工程上实现支持新组创建的异步模式。即组创建时，允许有成员不在线，但只要在合理的时间内上线，并发现自己被指定加入新组，则自行发起上述建组步骤的第 1，2 步，将自己的分享秘密分发给其他组员，其他组员收到它的分享秘密后，反馈自己的分享秘密给该组员。这样能异步收齐所有组员分发的分享秘密即可新组创建成功。

组动态增加组员

由于在 P2P 网络上有很多不可预测性，在工作组建组成功后的存续周期（ValidPeriod）内，节点下线或者不工作的情况会大概率发生。当不在线节点过多时，会直接造成(t,n)门限签名无法完成。另一方面，每个纪元（epoch）通过工作组的分红合约的记录，可以得知组内那些长期怠工的节点（可能下线，也可能恶意怠工）。所以我们定期剔除怠工节点后，要考虑组动态增加组员算法。

算法考虑的问题是：在初始组员n人，由于活跃度的原因，组成员变为只有$r(r \geq t)$个活跃用户时，如何为组成员添加新用户。

由于用户顺序对结果不影响，不妨假设该r个活跃用户为$(ID_1, ID_2, \cdots, ID_r)$。

令$F(x) = \sum_{i=1}^{n} f_i(x)$，在组创建中，我们证明了$F(x) = \sum_{i=1}^{r} S_i \prod_{j=1, j\neq i}^{r} \frac{x-ID_j}{ID_i-ID_j}$。对于新加组员$ID_{new}$，容易知：

$S_{new} = F(ID_{new}) = \sum_{i=1}^{r} S_i \prod_{j=1, j\neq i}^{r} \frac{ID_{new}-ID_j}{ID_i-ID_j}$是$t-1$次多项式$F(x)$上的一点，满足当组员人数超过门限值$t$时，可以通过各自的组分享秘密，恢复出组初始秘密$SK$。由上式可知，新组员的组分享秘密由当前组活跃的$r$个组员的组分享秘密线性构成。如何不泄露自己分享秘密S_i的前提下，让新组员ID_{new}构建出它自己的组分享秘密S_{new}？

具体解决流程如下：

1. 当前活跃的r个组员分别自行选取各自的秘密多项式$g_i(x) \in GF(p)[x]$，

   $$g_i(x) = a_{i,0} + a_{i,1}x + a_{i,2}x^2 + \cdots + a_{i,t-1}x^{t-1},\ i = 0,1,2\cdots,r$$

   其中多项式系数$a_{i,1}$，$a_{i,2}$，$\cdots, a_{i,t-1} \in GF(p)$，均为每个组员自取的随机数。且满足$a_{i,0} = g_i(0) = S_i \prod_{j=1, j\neq i}^{r} \frac{ID_{new}-ID_j}{ID_i-ID_j}$，$i = 0,1,2\cdots,r$。
2. 当前活跃的r个成员各自分别计算给其他组员的分享秘密并将该分享秘密发给对应的组员。即：第 i 个组员，计算$D_{i,j} = g_i(ID_j)$，发给第 j 个组员。其中$i = 0,1,2\cdots,r;\ j = 0,1,2\cdots,r;$
3. 当组员收集齐其他组员发给自己的数值后，计算所有收到的数值之和$D_i = \sum_{j=1}^{r} D_{i,j} = \sum_{j=1}^{r} g_j(ID_i)$，并将该数值发送给新组员。
4. 新组员计算$A(x) = \sum_{i=1}^{r} D_i \prod_{j=1, j\neq i}^{r} \frac{x-ID_j}{ID_i-ID_j}$，由以前证明知$A(x) = \sum_{i=1}^{r} g_i(x)$。则$A(0) = \sum_{i=1}^{r} g_i(0) = \sum_{i=1}^{r}(D_i \prod_{j=1, j\neq i}^{r} \frac{-ID_j}{ID_i-ID_j})$,新组员的组分享秘密$S_{new} = A(0)$。

此时，活跃的r个组员和新组员构成一个新的组，但老组员各自的组分享秘密没变，组的初始秘密SK也没变，当新组人数超过门限t 时，可以通过各自的分享秘密，恢复出组初始秘密SK。

正确性证明：

因为

$\sum_{i=1}^{r} D_i \prod_{j=1,j\neq i}^{r} \frac{x-ID_j}{ID_i-ID_j} = \sum_{i=1}^{r} \sum_{h=1}^{r} D_{h,i} \prod_{j=1,j\neq i}^{r} \frac{x-ID_j}{ID_i-ID_j} =$
$\sum_{h=1}^{r} \sum_{i=1}^{r} D_{h,i} \prod_{j=1,j\neq i}^{r} \frac{x-ID_j}{ID_i-ID_j} = \sum_{h=1}^{r} g_h(x)$,

所以

$\sum_{i=1}^{r} D_i \prod_{j=1,j\neq i}^{r} \frac{x-ID_j}{ID_i-ID_j} = \sum_{h=1}^{r} g_h(0) = S_{new} = F(ID_{new})$

安全性证明：

由于$F(x) = \sum_{i=1}^{n} f_i(x)$是最高次数为$k-1$次的多项式，多项式系数$k$个，而当$t < k$时，只能构建$t$个等式，无法求解出多项式系数（解不唯一），从而无法确定$F(x)$。所以等$t < k$时，无法得到组初始秘密SK的任何信息。

块链模型

数据区块的数据结构包含，但不限于以下信息：

- ✧ BlockHeader： block 块头信息
  - ▫ Hash： 当前块 hash
  - ▫ Height： 当前块高
  - ▫ CurTime: 当前块铸块时间
  - ▫ PreHash： 上一块 hash
  - ▫ PreTime： 上一块铸块时间
  - ▫ Castor: 提案人 ID
  - ▫ GroupId： 工作组 ID
  - ▫ Signature：组签名
  - ▫ Rand： 随机数
  - ▫ Transactions[]: 交易集 hash 列表

- Nonce

真随机数生成

结合 ECDLP 的 Shamir 秘密共享方案

我们采用的 Barreto-Naehrig 椭圆曲线$E: y^2 = x^3 + b,\ b \in GF(p)$

其中，有限素域$GF(p)$：

$$p = 36x^4 + 36x^3 + 24x^2 + 6x + 1$$
$$r = 36x^4 + 36x^3 + 18x^2 + 6x + 1$$

这里x是 63 bit 的数，p，r均为 bit 长度在 256 左右的素数。

两个有限群$G_1 = E\big(GF(p)\big)[r]$，$G_2 = E[r] \cap Ker(\pi_p - [p])$，

我们采用近年来论文中 bn 椭圆曲线上最优线性对算子[4] $e: G_1 \times G_2 \to GF(p^{12})$ 定义为：

$$e(Q, P) = (f_{6x+2,Q}(P) \cdot H)^{(p^{12}-1)/r}$$

这里$H = l_{Q_3,-Q_2}(P) \cdot l_{-Q_2+Q_3,Q_1}(P) \cdot l_{Q_1-Q_2+Q_3,[6x+2]Q}(P)$

$f_{6x+2,Q}(P)$是可以通过 Miller 算法计算的。

由双线性算子的特性：

对任意$P_1, P_2 \in G_1$，$Q \in G_2$，成立$e(P_1 + P_2, Q) = e(P_1, Q)e(P_2, Q)$

对任意$P \in G_1$，$Q_1, Q_2 \in G_2$，成立$e(P, Q_1 + Q_2) = e(P, Q_1)e(P, Q_2)$

对任意$p \in G_1$，$Q \in G_2$，$a, b \in \mathbb{Z}$，成立$e([a]P, [b]Q) = e(P \cdot Q)^{ab}$

这里，记$[\cdot]$为椭圆曲线上的倍乘。如$[a]P$是椭圆曲线上点P的a倍乘积。

基于椭圆曲线的签名算法：

- m ∈ {0,1}*: 需要签名的消息（二进制表示）
- 计算$R = H(m) \in G_1$

- 计算$\sigma = [x]R$，其中x是用户签名私钥，σ即为所得签名

那么结合上述的组创建（去中心化 Shamir 秘密共享），对组员签名私钥$\{S_i\}$与组私钥SK，存在如下关系：

$$SK = \sum_{i=1}^{k} S_i \prod_{j=1, j\neq i}^{k} \frac{-ID_j}{ID_i - ID_j}, \quad k \geq t$$

则对消息$\mathrm{m} \in \{0,1\}^*$，每位组员的签名为$V_i = [S_i]R$，由双线性椭圆曲线的上述特性，可得：

$$[SK]R = \left[\sum_{i=1}^{k} S_i \prod_{j=1, j\neq i}^{k} \frac{-ID_j}{ID_i - ID_j}\right] R = \sum_{i=1}^{k} \prod_{j=1, j\neq i}^{k} \frac{-ID_j}{ID_i - ID_j} V_i$$

简而言之就是，在双线性椭圆曲线上，上述的构组方式所得组员签名私钥对消息签名后，当得到$k \geq t$条组内成员的消息签名，可以用 Lagrange 插值多项式得到组私钥SK对该消息的签名。

在 Shamir 秘密共享算法中，恢复组私钥SK，是要泄露$\{S_i\}$的。利用双线性映射 e 的性质，在不泄露$\{S_i\}$的情况下来完成组私钥SK的签名，保证了组员签名私钥可以不断复用。通过使用该技术，可以通过门限签名来实现组内共识，而且效率比拜占庭算法（BFT）更高。

VRF 随机数生成方法

由于组私钥SK无人知晓，所以该签名$[SK]R$具有不可选择，不可预测，不可改变，却可以通过组公钥gpk来验证签名$[SK]R$是否由该组签出的。它是一种 VRF 随机数生成方法。

SPOW 系统采用的随机数生成方法，就是采用上述组协作的 VRF 方法。

记$B^i.Rand$是第i轮 slot 出的区块的 Rand 值。对于第r轮 slot，使用的随机数

$$\mathrm{R}_r = hash(B^{r-1}.Rand)$$

按选组策略可以确定第r轮 slot 的工作组，由当前工作组对$(r|R_r)$做签名，收

集组内门限t个以上的签名后，恢复的组私钥SK对$(r|R_r)$签名作为当前 slot 生成随机数，写入第r轮 slot 所出的区块的$B^r.Rand$。

$$B^r.Rand = recover(sig_1(r|R_r), sig_2(r|R_r), \cdots, sig_t(r|R_r))$$

前一块的 Rand 确定当前块的工作组，当前工作组以上述公式产生当前块的 Rand，确定下一块的工作组。其中R_0是创世块中的 Rand，由系统初始设定。

若第r轮 slot 的工作组没能完成出块，则第$r+1$轮使用的随机数

$$\mathrm{R}_{r+1} = hash(hash(B^{r-1}.Rand))$$

按选组策略可以确定第$r+1$轮 slot 的工作组，对$(r+1|\mathrm{R}_{r+1})$做门限签名，产生当前块的 Rand。若第$r+1$轮仍未出块，当前 slot 使用的随机数，以及当前块随机数生成准则，以此类推。

选组策略

假设当前进入第r轮 slot，计算：

当前纪元$e = r/epoSlots$

随机数$\mathrm{R}_r = hash(B^{r-1}.Rand)$,

从组链中获取当前存续的工作组列表$\{\mathrm{g}B\}$

$$\{\mathrm{g}B^{\mathrm{i}} | \mathrm{g}B^{\mathrm{i}}.\mathrm{GIS.Activation} \le \mathrm{e} < \mathrm{g}B^{\mathrm{i}}.\mathrm{GIS.Deactivation}\}$$

以该随机数R_r作为随机种子，用伪随机数生成函数 PRG 可以随机在上述存续工作组列表$\{\mathrm{g}B\}$里确定当前工作组。这些选择都是其他节点可审计验证。

后期我们可能会考虑拿组的健康指数(组员健康指数和)，利用追随中本聪算法（FTS）来作为选组的依据。

组工作流程

考虑某个工作组 G 的工作流程，如下图所示

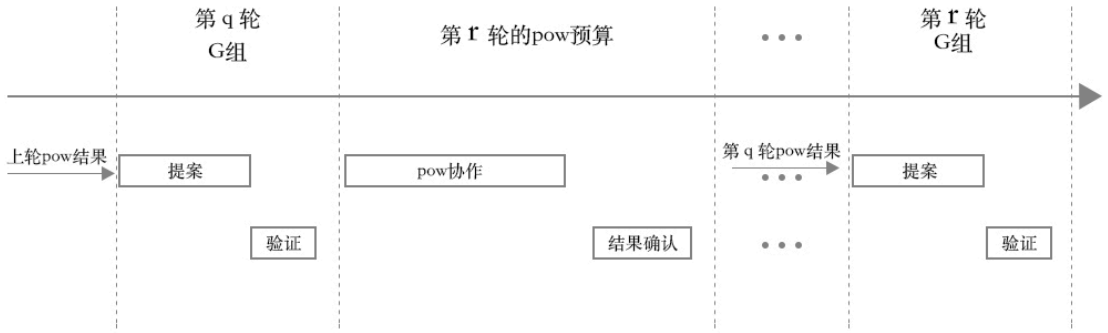

图 5. 组内工作流程图

假设第 q 轮和第 r 轮 G 组均被 VRF 选中为该轮铸块组，其中 r>q，G 组工作流程大概如下：

1. 以上一轮组内 POW 预算结果为第 q 轮提案并通过组内验证，完成第 q 轮铸块；
2. 组内所有成员进行算力协作并通过组内确认后生成第 r 轮的 POW 结果；
3. 第 r 轮到来后以步骤 2 的 POW 预算结果为本轮提案并通过组内验证，完成第 r 轮铸块；
4. 转到步骤 2；

每一轮工作组对区块提案完成组内验证后即可全网广播，我们优先将区块广播到下一轮的工作组，在底层经过组通讯优化的 p2p 网络中，此机制使得下一轮工作组可以更快为该轮铸块准备，降低出块延时，同时也降低了 DDOS 风险。

POW 预算

传统 POW 算法需要全网节点对每一轮铸块进行实时 POW 计算，这是导致出块延时长性能低下的根本原因。

SPOW 算法采用分组记账组间通过 VRF 随机轮换的机制，使得每个组在两轮铸块之间有很多的算力空闲时间，若充分利用这些空闲算力提前去做下一轮

POW 计算，将能大大降低出块延时提高系统性能。基于这一点，我们引入了 POW 预算机制。即每一轮的工作组基于当前区块为下一轮的 POW 结果进行预算，在下一轮到来之后可直接以该预算结果为本轮区块提案。

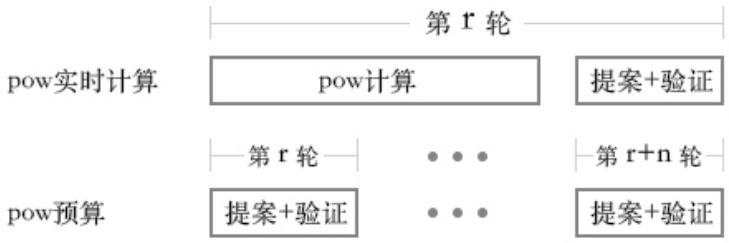

图 6. POW 预算和实时计算比较

如图所示，假设 POW 计算耗时为t_{pow}，提案并验证耗时为t_{pv}，则

- POW 实时计算下，每块延时为$rt_{real} = t_{pow} + t_{pv}$；
- POW 预算下，每块延时为$rt_{pre} = t_{pv}$；

令$t_{pow} = \mathrm{n} * t_{pv}$，则$rt_{pre} = \frac{1}{n+1} rt_{real}$。

直观上，POW 预算机制下，出块延时等于组内提案并验证的耗时，这个耗时仅为实时计算的$\frac{1}{n+1}$，而 n 取决于 POW 计算的耗时和组内提案验证的耗时比。因此，通过 POW 预算机制，TAS 达到了 POS 系统的性能。

从宏观上看，TAS 系统就像是包含了多个独立运行却又通过链互相耦合的协作型比特币系统，拥有低能耗的同时具有 POS 系统的高性能。

POW 协作

在每一轮的 POW 预算中，每个组内节点需要在ΔT时间内，根据本组当前铸出的区块信息加上一个随机数 nonce 构建一个 128bit 的整数 m，使得对 m 的双

重 SHA256 运算结果落在指定的区间，区间的大小表征了本次计算的难度：

$$Difficult \geq SHA256\big(SHA256(m)\big)$$

计算出满足上式条件的节点将其随机数 nonce 在组内广播，当收齐组内 k 个成员的 nonce 值后，组内成员将对此 k 个成员的 nonce 序列按照其 POW 结果难度排序，并计算总难度值，然后进行签名，该 k 个成员在下一轮出块时按照 nonce 序列的顺序以总难度值作为区块难度进行提案。由于 k 个成员的 POW 计算共同为区块难度值产生贡献，因此无论谁的提案最终胜出，k 个成员均可享受铸块收益。

本质上，组内成员在进行 POW 协作而非竞争。首先，只要节点参与 POW 计算，即使是拥有弱算力的节点也有可能得到铸块收益，这提高普通节点积极参与 POW 计算的积极性，同时，拥有强算力的节点由于能算出更大难度值而获取更多铸块收益，这保证了系统的公平性。其次，每个工作组从 POW 预算，到区块提案，再到区块验证，每个阶段每个节点都是协作关系，从整体上保证所有组员利益的一致性。最后，POW 协作机制保证了 TAS 系统全程无单点的特性，这使得 TAS 系统在抗 DDOS 攻击上有更强的安全性，详见安全分析章节。

综上所述，TAS 系统的 POW 协作机制，既同时兼顾了大众化和差异化，提高了安全性，又使得组员间能互相监督互相促进，这将有利于促进 TAS 系统生态的良性循环。

组外验证

组外节点收到新块的消息，首先判断该块的验证组合法性，然后从块内容中获得组 ID，然后通过组链上该组的公钥，以此来验证块上的组签名是否正确。

若签名验证正确，本地上链成功，则将该块继续对外广播，否则停止对外广播。

通讯优化

参考比特币的 P2P 网络的传播性能：1KB 消息，在 1 秒钟内完成全网 95%的传播，而 1MB 消息需要 1.5 分钟完成全网 95%的传播。我们考虑到组成员散布在世界各地，而且工作组会以多个区块的方式提出候选区块，所以必须对通讯做相应的优化。我们考虑采用组内以 Block header 传给验证组。组内对 block header 的的 Hash 达成强一致，仅仅保证了 block 内容以及时序。用户账户是否存在 double-spending 没法验证。所以我们是在快速对 hash 达成一致后，等交易同步到本地后，由上链时的账户状态验证来保证交易的有效性。这样可以达到更高的共识效率。如果提案矿工是诚实的，这样的流程效率比传统的以整个 block（包含区块内所有交易内容）为消息传输的方式要高。如果提案矿工是恶意的，这区块在上链时会验证失败，由上述的组外验证保证不会被传播，这样也保证了安全。另外，如果提案矿工是恶意的，该块的验证组验证后会发起该块的分红合约，但上链验证失败使该区块上链失败。块链中分红合约与块链的不一致，可以作为作恶凭证，对作恶的提案矿工给出相应的惩罚。比如降低出块矿工的健康指数，保证金扣除等。

分叉处理

分叉选择：

由于共识机制的特性和 P2P 网络状况的不确定性，软分叉在区块链中无法避免，在出现软分叉之后，TAS 链的各节点以总算力难度最高的分叉

$totalD = \sum D_i$，其中D_i是各个区块的算力难度。

作为规范链，分叉节点会通过分叉调整简单快速地将本地链调整至规范链，从而保证链的一致性。

分叉调整：

节点遇到分叉之后，首先找到分叉点，然后比较分叉点到当前高度的总难度值做出选择。在寻找分叉点的过程中，采取局部比较的方式，每次请求一定步长的目标链片段进行二分比较，然后根据比较结果调整步长，从而快速定位到分叉点进行分叉调整。

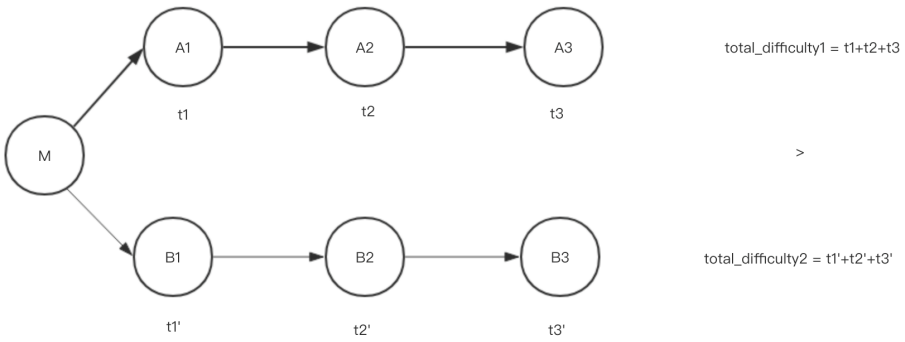

图 7. SPOW 分叉处理

共识分析

去中心化分析

POW 共识是高度去中心化的算法，但是随着矿机，矿池出现，它们成为 POW 共识的隐性中心。用户使用普通算力设备参与铸块，几乎无法收益。SPOW 共识，对算力设备随机分组，具体轮次由真随机数确定工作组。简单来说，POW 共识是由全网算力最强者胜出，SPOW 共识是由随机指定的工作组中的算力排名前若干名协作胜出。SPOW 共识是强调组内协作方式，所以奖励会有多名参与者获得。SPOW 共识采用分组方式，让更多的铸块矿工获得系统奖励。另外，SPOW 的工

作组还会定期重建，这样使得分组结果更多样化，让更多的参与者能获得系统奖励，解决了算力隐性中心的问题。同时因为分组进行 POW 计算，也大大降低了全网的能耗浪费。

SPOW 共识还对参与验证的组成员进行奖励。用户可以使用普通设备参与验证奖励。每轮工作组的选择也是由真随机数确定的，所以每个工作组都是有相同概率获得验证机会。

安全分析--攻击防御

无利害攻击(nothing at stake)

在纯 POS 算法中，只为创造区块提供奖励，恶意出块或者基于错误的分叉出块都没有惩罚措施，这就出现在多链竞争条件下，理智的矿工的最佳策略是在每条链上进行”挖矿”。这意味着在该机制下，不管哪条链胜出，矿工都会得到奖励,由于他们没有花费物质上的算力，所以矿工以很低的成本就可以使得自身利益最大化。

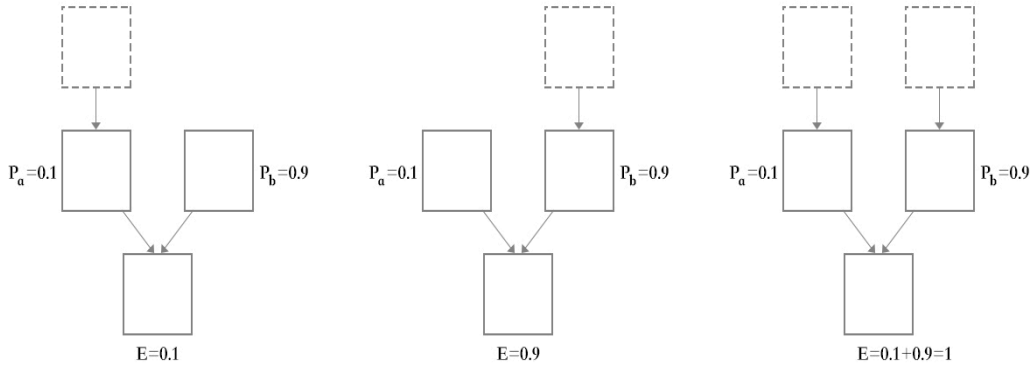

图 8. 无利害攻击示意图

如图所示, P_a,P_b 分别为两个分叉的胜出概率，当矿工在两个分叉上均投票时，E(收益期望)得到最大化. 在极端情况下，全网矿工唯利是图，即使没有攻击者，也有可能无法达成共识。

在传统的 POW 算法中，所有矿工进行实时算力竞赛，要实施无利害攻击必须分散算力到两个分叉中，这样会大大降低自己的竞争力，因此传统 POW 天然抗无利害攻击。

与传统 POW 不同，在 SPOW 算法中，每组矿工的算力竞赛在被 VRF 选中前就已提前完成，意味着每组矿工在被 VRF 选中后，只需要把提前算好的区块难度打进区块并完成组签名即可出块，表面上看，在这种机制下实施无利害攻击成本变得很低。

无利害攻击的原因之一在于，每个矿工在一段时间内有权对所有区块进行投票，意味着分叉情况下，矿工有经济动力保留每条分叉，并在上面投票，最终导致链无法收敛。实际上，在 SPOW 体系中，VRF 决定了每一轮在多条分叉上具有出块权的组只有一个，同时收益也只有一份，剩余无权出块的组均没有经济动力保留两条分叉，他们会在两条分叉中作出选择并最终达成一致。

无利害攻击的另一个原因是攻击者在两条分叉上投票成本很低又能确保收益的最大化。在 SPOW 体系中，出块组在完成本轮铸块后，必须在规定时间内预算下一轮的难度解，若在两个分叉上出块，则必须分散算力去做下轮的 POW 计算，算力的分散意味着求得更优解的难度增加，从而直接影响下一轮矿工的收益。

基于以上两点，我们认为 SPOW 算法机制能天然的抵抗无利害攻击。

长程攻击(Long-range attack)

在 POS 算法中，出块的速度没有限制，在系统初期，矿工不多，如果这些矿工联合起来，回到系统初期的状态开始铸块，在短时间内铸出一条更长的链，这时候用户无法辨认哪条链是主链，甚至攻击者发布的链有可能战胜主链。当前很多 POS 实现是通过限定区块能回滚的数量加大攻击的难度。

在 TAS 系统中，所有区块难度设定了上下限，记为D_{max}和D_{min}，并且有$D_{max} = k \cdot D_{min}$，每个 epoch 宽度为$H_{epoch}$，每个工作组的工作周期为 n 个 epoch，工作周期过后组会自动解散进入重建流程。假设攻击者选择回退 H 块实施长程攻击，并且攻击者控制的组占比为 x，那么在攻击过程中，由于 VRF 的不可选择性，攻击者每次只能在被选中的时间窗口出块，而未被选中的时间窗口出空块。长期来看，攻击者链的有效区块比例与攻击者组占比正相关。n 个 epoch 后链状态如下图：

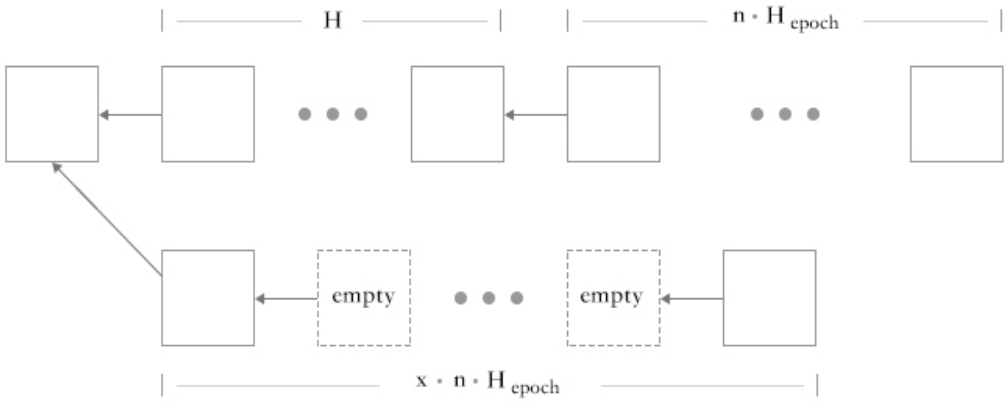

图 9. 长程攻击示意图

攻击者在工作周期内能算出的最大链累积难度为：

$$D_{attack} = x \cdot n \cdot H_{epoch} \cdot D_{max} = x \cdot n \cdot H_{epoch} \cdot k \cdot D_{min}$$

此时主链累积的最低难度为

$$D_{main} = \left(H + n \cdot H_{epoch}\right) \cdot D_{min}$$

攻击者得逞的条件为

$$D_{attack} > D_{main}$$

简化后得

$$n \cdot H_{epoch} \cdot (k \cdot x - 1) > H$$

由此可见，攻击者只能回退到有限高度去实施长程攻击。尤其是，当 k·x<=1 时，攻击成功的概略将为 0。意味着攻击者必须控制更多的组来降低攻击的难度。下表给出不同 k 值条件下，攻击者至少需要控制组的比例：

k 取值	x 范围
1	无解
2	大于 50%
3	大于 33%
5	大于 20%

事实上，上述推论成立的条件非常苛刻。首先，攻击者控制的所有组必须处在相同的存续周期中，这样才能保持攻击者的攻击力，但这在随机建组模型上，攻击者很难实现。其次，推论假设攻击者每次出块都能算出区块难度上限，并且诚实者都只算出难度下限，只要两个条件的一个不满足，都将大大减小 H 可选的范围。最后，诚实者会拒绝接受过期（n 个区块以前）的区块，即系统限制了能回滚的区块数量。

简而言之，难度区间的设定和区块时效限制了系统能回滚的块数量；随机分组策略和组定期重建机制使得攻击者无法长期保持高强度攻击，因此长程攻击在 TAS 系统变得几乎不可能。

DDOS 攻击

DDOS 攻击主要是攻击者通过组织一批计算机，对服务节点发起大量的请求，从而使得服务节点忙于处理请求而耗尽资源，无法继续对外提供服务的攻击方式。比特币和以太坊全网节点进行 POW 计算，没有任何中心节点，因此天然的抗 DDOS。我们认为去中心化程度将决定抗 DDOS 的安全程度。SPOW 算法通过 VRF 在全网工作组中随机选择每一轮的出块组，表面上出块组成为了某一刻系统的中心，实际上由于 VRF 的随机不可预测性，这个中心每一轮都在变化，这是一个动态变化的中心，本质上也是去中心化的。

对 TAS 的 DDOS 攻击分为两类，一类是组外 DDOS，即组外节点企图攻击某一时刻的出块组，致其不能出块。此类攻击者必须做到以下四点:

1) 在每一轮开始的时候实时计算得知当前时刻的出块组
2) 通过查询 TAS 的底层网络，得到组内每个成员的网络地址
3) 组织足够多的 DDOS 请求，发给组内超过 51%的成员，致使他们瘫痪
4) 持续的执行上述 3 个步骤

前两点很容易实现；

对于 3），我们认为 DDOS 攻击一个节点最有效的方法是将其带宽打满。假设运行 TAS 的矿工平均带宽为 10Mbps(这个需求在现代很容易满足)，则要打满 100 个节点的带宽，攻击者的带宽至少是 10 * 100 = 1Gbps，再加上相应的机器电力成本，这个代价已经不是普通人能够承受的。更重要的是，出块组和攻击者几乎同时开始工作，而且出块组的时间窗口很短，以至于在 DDOS 到来前出块组有很大概率已将块广播出去。

对于 4），即使攻击者在本轮攻击成功,当前组无法出块,当该组的时间窗口过后，下一个组自动开始下一轮的出块，以此类推。攻击者必须持续高强度的发送 DDOS 请求给不同的目标机器。我们认为 DDOS 攻击对静态稳定的中心节点具有很大威胁，而对迅速变化的中心节点则威胁系数急剧下降，而且攻击成本更高。

另外一类 DDOS 攻击是组内 DDOS。由于最终出块收益与算力直接相关，使得组内算力小的节点有动力攻击算力大的节点，致其无法出块，从而获得更多出块收益。SPOW 算法规定每个节点的计算难度定在某个固定区间，保证算力不同的节点收益有所差异却又不会太大，这样保证公平性。由于 DDOS 成本非常大，因此在这种机制下，组内节点发起 DDOS 攻击的成本与攻击带来的收益远不成比例，攻击者得不偿失。同时，TAS 的组内 POW 计算并非节点竞争，而是协作，每个节点对出块的难度贡献算力，最终无论谁出块，参与贡献的节点均有收益，因此，算力小的节点也没有必要攻击算力比其大的节点。

综上所述，VRF 的随机不可预测保证了 TAS 系统的去中心化特性，组内算力协作共享收益机制使得组内成员趋向互相合作，这两点保证 TAS 系统抗 DDOS 攻击。

女巫攻击(sybil attack)

女巫攻击的一种形式是攻击者注册大量的账号，并且使用这些账号进行挖矿，以提高自己的收益，尤其是在 POS 系统中,账号的注册成本非常低，而且不需要很多的算力，只要缴纳押金即可参与，从而有钱的人就有更大的话语权，而变得更有钱，这破坏了系统的公平性，也容易导致中心化.

TAS 仍然沿用缴纳押金的方式参与出块，同时也引入设备即押金的设计，即设备健康度更高的节点可以缴纳更少的押金达到同样的效果。设备健康度越高或者缴纳押金越多的节点被选中分组的概率更高。这意味着攻击者若想提高被选中的概率，则必须遵循 TAS 的规则，积极参与出块以提高其设备的健康度，或者缴纳更高的押金达到同样的目的。任何被分组选中的账号，可以通过 POW 计算或者参与区块验证获得收益，同时为设备累积更高的健康指数。相反，任何既不参与 POW 计算也不参与验证的账号，会越来越难被分组选中。

另外一种攻击形式是攻击者围绕在诚实者周围，通过一定手段与诚实者交互企图窃取利益. 在 TAS 系统中，节点之间交互类型主要有三种，第一种是单向交互，即接收者接收消息后通过合法性验证并进行存储即可，如数据同步消息，由于单向交互性使得攻击者基于此种交互并不能获取任何利益。第二种是双向交互，即接收者接收消息并通过合法性验证后向发送者(或组内其他成员)进行相关反馈，如块验证消息，建组消息；攻击者可以试图伪装成组员和被攻击者进行建组流程或者共同参与块验证，企图获取利益。由于 TAS 的建组流程是由指定的父亲组发起的，建组的成员经过父亲组门限值以上的节点签名，具有不可篡改，可审计的特性，因此被攻击者可以轻易的判断与之交互的节点是否属于组内成员。第三种是链式交互，即接收者接收消息后只做简单的转发。显然这种简单转发不会让攻击者收益。最后值得强调的是，上述攻击无效的前提是矿工的私钥不泄露，TAS 系统所有的交互都不涉及私钥的传递。

综上所述，TAS 系统采用铸块参与度与设备健康指数相互反馈相互促进的机制使得攻击者无法作恶。安全可验证的消息通讯机制使得攻击者无法通过任何手段窃取被攻击者的利益。

最后操作者攻击

TAS 系统中所有节点通过分组协作方式参与记账，每个区块由组内门限个以上成员签名，组外节点可通过组公钥进行验证，因此具有不可篡改的特性。最后操作者若企图修改数据，则会导致最终校验失败，攻击者无利可图。

另外，最后操作者选择不作为也是攻击的一种手段。然而，在 TAS 系统中，任何组内节点都可以是最后操作者，不存在单点，只要有诚实节点，攻击者就不会得逞。极端情况下，攻击者控制了某个组所有成员，在最后出块时，选择不出，则系统会跳过本轮继续下一轮。攻击者同样无利可图。

双花攻击

双花攻击的常见手段是制造另一个能战胜主链的分叉。攻击者首先需要在某个组中控制超过一半以上的节点，并在轮到攻击者所控制的组出块时，提前将自己的一笔交易广播出去，同时自己也开始在另一个分叉铸块并把该交易打进块中，等待该笔交易在主链中被确认(TAS 的确认时间很短)后，再把自己的分叉广播出去，若攻击者的分叉链难度大于当前主链新增的累积难度，就会被矿工作为新的主链，如此一来，先前的交易就像不存在一样，以此达到攻击目的。此类攻击者的攻击策略和长程攻击类似，具体分析请参考长程攻击防御章节。

私自挖矿攻击

在比特币系统中，由于所有矿工可以在每个高度上进行 POW 挖矿，因此算力强的节点可以在挖到某块后，先藏着，并私自开始下一轮挖矿，以此获得先机。

TAS 系统中所有矿工通过分组协作方式进行挖矿，每个矿工挖出的块必须经

过组内大多数矿工的签名才能广播出去，因此私自挖矿对单个矿工并不成立。而组间的轮换完全由 VRF 决定，私自挖矿并不能使组获得下一轮挖矿权，因此私自挖矿对组亦不成立。

51%攻击

在纯 POW 的系统中, 51%攻击是致命的不可恢复的。因为攻击者拥有足够的算力挖出一条难度更大的更长的链去淘汰主链，而诚实者会渐渐接纳攻击者链，攻击者永久胜出。

TAS 系统将矿工随机分配到不同的组, 51%攻击可能导致攻击者控制了绝大部分工作组。在这种情况下，主链将缓慢延长，攻击者链由于拥有更多的组而延长速度更快，但是诚实者会拒绝接受攻击者链，他们会一直坚持延长主链。此时绝大部分用户能轻易感知系统存在两条不同的链而不敢轻易交易, 从而导致攻击者收益甚微，这与其发动 51%攻击的成本完全不成比例。

如果一个普通用户随机连接的所有节点都被攻击者控制，则此用户由于无法感知异常而继续信任网络，我们认为这个用户是能给攻击者带来收益的，暂且称此类用户为受害用户。显然受害用户越多，攻击者收益越大。下面简单分析一下攻击者和受害用户的比例关系。

假设每个用户都随机连接的 n 个节点中有 90%的节点都被攻击者控制，则该用户成为受害用户。假设全网节点数 W，攻击者控制比例为 x，则受害者比例为：

$$\mathrm{v} = \frac{\sum_{i \geq 0.9n}^{n} \binom{W \cdot x}{i} \binom{W \cdot (1-x)}{n-i}}{\binom{W}{n}} i \in N$$

更普遍的，设W=10000，n=10，v 和 x 的曲线如下图：

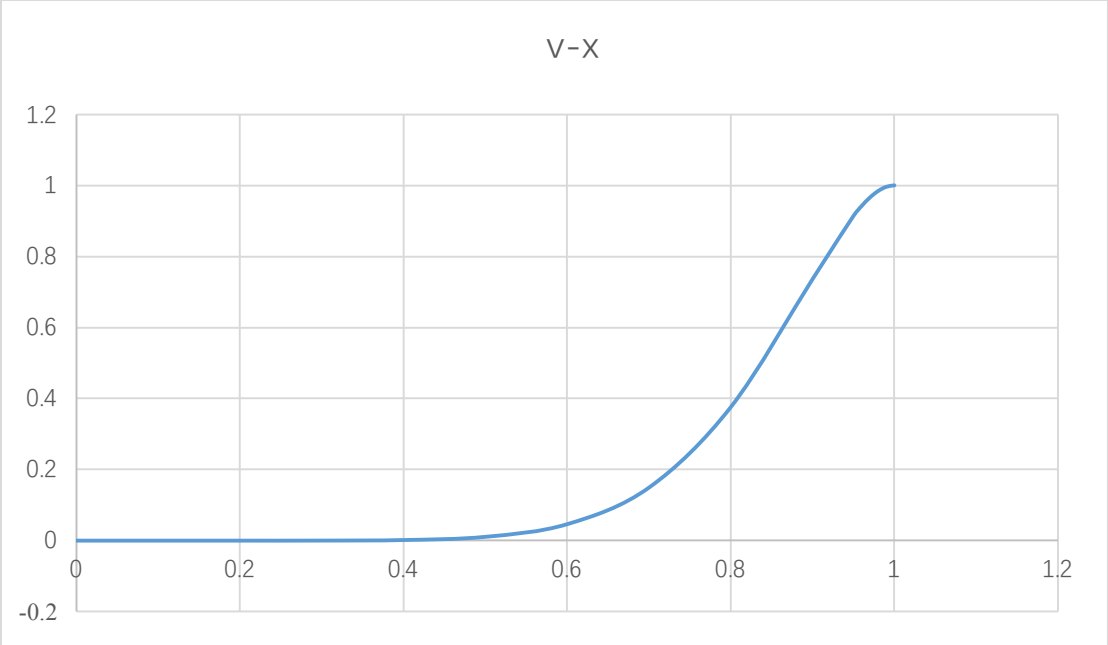

图 10. 攻击者比例和受害者比例关系图

取几个特殊点得如下表格：

攻击者比例	受害者比例
0.5	0.01071
0.8	0.3757
0.9	0.7361
0.95	0.914
1	1

由此可知，当攻击者控制 50%的节点时，受害者比例仅为 1.07%；而理论上只有当攻击者控制了 100%的节点时，攻击者才可以控制整个网络。如果受害者比例超过 90%后，攻击者就有利可图，那么攻击者需要控制 95%以上的节点。这个攻击的成本非常巨大。相反，当攻击者由于无利可图而渐渐退去，诚实者比例大于 95%后，系统就能恢复正常。

综上所述并结合于目前为止在 POW 类系统中发生的多次 51%攻击案例，我们认为 TAS 系统能抗 95%甚至更高的攻击（在系统冷启动和低算力状态进一步提高系统健康度阈值），因而具有更高的安全级别。

P2P 网络

TAS P2P 介绍

我为人人，人人为我，P2P 和基于 P2P 发展起来的 DHT/KAD 等技术是去中心化自治的早期雏形，也出现了不少互联网时代的成功案例。

区块链推动了 P2P 技术的进一步发展，从分布式存储到共享计算和带宽，以及和密码学的有机结合，都对 P2P 技术提出了更高的要求。我们观察到目前大部分公链（POW/POS/DPOS）使用的仍是较早的 P2P 技术，比如不支持局域网穿透，需要搭建静态 IP 专用网络或者手动开启路由设备的 UPNP 才能加入到 P2P 网络中。静态 IP 的网络费用高昂，而运营商提供的大部分入网设备需要破解才能开启 UPNP 功能，普通节点想要加入到区块链 P2P 网络的门槛仍然非常高。同时比特币和以太坊的节点网络，越来越被矿机和矿池垄断，普通用户无论从物理网络层面和逻辑共识层面都越来越难以成为铸块矿工并拿到合理的收益。TAS 革命性的提出了 SPOW 共识机制，让笔记本、手机（轻节点）和高性能 PC、矿机（重节点）都能参与到铸块共识，为了达到这个目标，TAS 对传统的 P2P 技术做了较大的技术升级。主要包括：

- 使用新型的 NAT 穿透技术（专利申请中）大幅提高节点在线率。
- 针对 SPOW 以组为单位铸块的模式，设计了 2 层 KAD 网络提高组内通讯效率。

- 使用 RUDP 代替 TCP，通讯延时降低 35%左右。

TAS P2P 架构

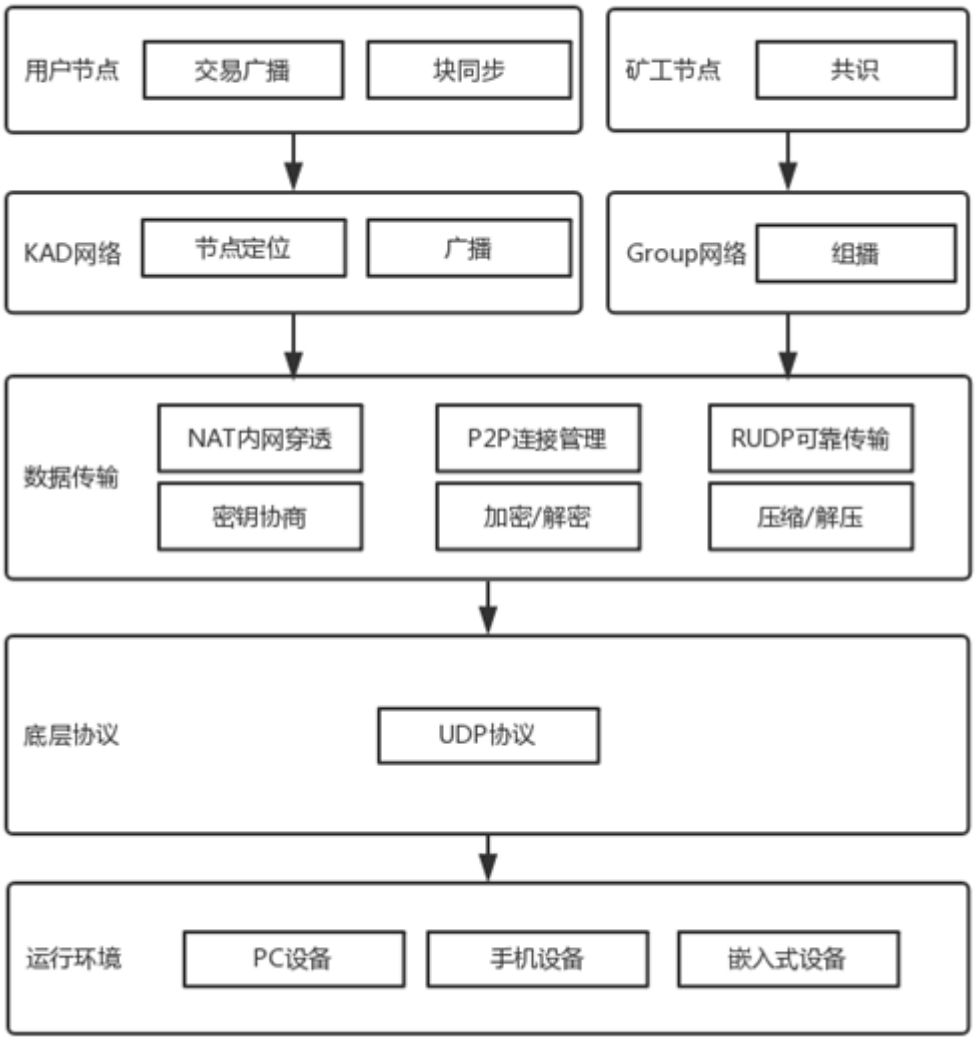

图 11. TAS P2P 架构图

NAT 穿透

P2P 网络作为区块链的基础设施，一定程度上决定了区块链能走多远。考虑互联网从 IPv4 全面升级到 IPv6 的长周期性，内网穿透仍是 P2P 的最重要技术之一。对于结合了广度和深度的 SPOW 共识机制，高穿透率保证了更多的矿工可

以自由的加入到节点网络，而参与的节点越多，越满足好人多于坏人这个区块链的基本理念。

主流的 NAT 穿透使用标准 STUN[13]作为解决方案，穿透率在 30%左右。STUN 将 NAT 设备分为四类，全锥型、限制锥型、端口限制锥型、对称型。我们对中国六万个网吧 NAT 设备进行了统计，其中全锥型占比 5%，限制锥型占比 7%，端口限制锥型占比 58%，对称型占比 30%，得到的理论穿透率为 56%（5% x 100% + 7% x 100% + 58% x 70% + 30% x 12% = 56.20%）。另外，由于 NAT 防火墙的存在，被动的 UDP 包到达 NAT 设备时，在连接跟踪模块会产生相应的记录，引起的副作用会导致随后的端口预测失败，进而导致整个穿透流程的失败。STUN 并没有考虑 NAT 防火墙的因素，按标准实现的穿透率仅在 30%左右。

NAT 穿透的本质是预测地址转换设备的映射端口，考虑到非对称设备可以准确的预测映射端口，所以对称设备如何处理，是决定穿透率高低的关键因素。STUN 是针对 90 年代以 ASIC 方式实现的路由设备进行设计的，而路由设备发展到今天，大多数路由设备已经升级到网络处理器和 linux 内核相结合的软硬件解决方案。通过对 linux 内核协议栈的分析和多年的工程实践，我们从端口映射规律重新定义了 NAT 设备类型，并将之分为三类：主机端口、固定端口和对称端口。用这种分类法重新对六万个 NAT 设备进行统计，其中主机端口占比为 75%，固定端口为 23%，对称端口为 2%，得到的理论穿透率为 96%（75% x 98% + 23% x 98% + 2% x 0% = 96.04%）。另外，路由器防火墙在接收到无相关性的被动数据包后会更改映射端口而导致随后的穿透失败，这是实际穿透率无法达到理论值的最大影响因素。TAS P2P 设计了 TTL 动态调整算法，针对不同节点间的距离动态调节穿透协议包的 TTL 值，使这个数据包既能通过本地的 NAT 设备，又不会在对方的

NAT 设备留下痕迹而影响映射端口，最终使实际穿透率基本达到了理论值的水平。TAS 通过多年在音乐下载、网吧多线路融合等 P2P 领域的技术沉淀和创新，NAT 设备 96%的高穿透率保证了大多数节点可以参与到铸块共识中，为上层 SPOW 提供了可靠的节点物理层高连通性。

内网穿透	STUN	TAS P2P
原理	RFC3489	Linux 内核协议栈
穿透率	30%	96%
设备分类	全锥型/限制锥型/端口限制锥型/对称型	主机端口/固定端口/对称端口
防火墙穿透	不支持	支持

图 12. 内网穿透技术对比

组播网络

TAS 的所有节点都会加入到全局的 KAD 网络中，KAD 网络采用基于汉明距离的拓扑方法来快速定位节点，该方法的好处是在定位过程中通过对数级的收敛保证了高效性。

TAS 的 KAD 网络启动后，某个节点都将和 16-32 个邻居节点建立连接，由邻居节点间的通讯完成交易广播、块链同步和组链同步。

针对分组的 SPOW 共识机制，TAS 构建了二层子 KAD 网络定位组成员节点。每个组在完成初始化后，在组成员之间建立组播网络。每个成员启动后会和 8 个同组的成员建立连接，来保证组内成员间铸块和验证的高效通讯。相比全局的

KAD 网络，二层组播网络保证了组内消息更快速的投递，同时减轻了对整个节点网络的负载。

RUDP

对于高在线率节点网络和大量碎片验证数据交互的场景，RUDP 相比 TCP 有着明显的优势。宏观来看，用 RUDP 代替 TCP 已是工业界的趋势，如谷歌提出的 QUIC 框架可以看做 RUDP 的超集，考虑到 TAS 要求的高连通性和组内协作铸块模式，TAS 在通讯层用开源且成熟的 RUDP 代替 TCP。

底层协议	TCP	RUDP
内网节点穿透	难	易
高质量网络传输速度	高	高
中质量网络传输速度	中	高（ARQ 快速重传）
低质量网络传输速度	低	高（FEC 冗余传输）

图 13. 通讯协议对比

分片并行计算框架

目前区块链系统在保证去中心化和高安全的前提下，普遍性能较差，所有的公链要支持大型商用 DAPP 落地都面临着扩容的问题，而扩容的核心是提高交易的吞吐量。为了更好的解决扩容问题，TAS 借鉴 Google 的 map-reduce 和阿里云批量计算的设计思想，设计了分片并行交易执行框架。

在我们的框架中，我们将组逻辑上分为计算组和出块组（组概念与构建流程参见组链模型）。也就是说，在一个铸块周期里，根据上一区块产生的随机数可

以选定了一个出块组，而根据签名与交易发起人的地址的运算，选定的组称之为计算组。计算组负责执行交易，出块组负责块的最终生成。计算组与出块组的输出都需要达成组内共识来验证正确性。

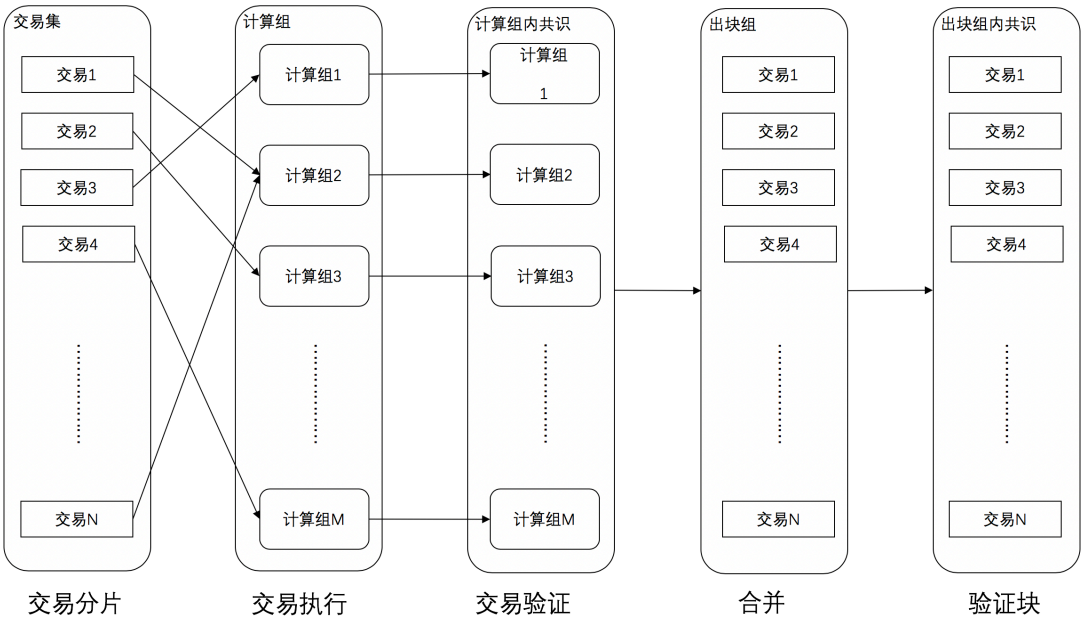

图 14. 分片计算流程图

交易分片

根据交易发起人的地址和上一块的签名，将交易发送到不同的计算组。在这里，我们并没有限制交易目标账户地址，比起状态通道模式具有更好的灵活性。相同的交易发起人发起的多笔交易，都将落到同一个计算组，用最小时间窗口即可验证双花问题。同时在分发交易的策略中采用了 SPOW 共识的 VRF 输出随机选择计算组，通过计算组复用和计算组的不可预测提高了系统的健壮性和安全性。

执行交易

计算组会根据当前账户状态执行交易，同时记录账户状态变化等情况。计算

组内采用 POW 竞争计算的方式选择哪一个节点将交易执行结果提交到验证组。

对于交易的执行往往需要依赖状态数据，例如账户余额、合约内保存的用户数据等。在我们的设计中，计算组保存了状态信息，一旦接受到一个新块，就会更新本地状态。所以计算组本地即可完成交易的执行，无需进行跨链等复杂交互。

交易验证

计算组内验证交易执行结果，使用门限签名方式生成组签名，提交出块组。

数据合并

出块组合并收到的账户状态变化，生成最终的账户状态，出块。

铸块

组内验证交易执行结果，生成组签名，最终铸块上链，参见 SPOW 共识机制的验证组共识出块部分。

智能合约

在 DAPP 快速发展的过程中，正面临着和传统 APP 发展一样甚至更多的由去中心化带来的问题。TAS 希望在保证契约性、安全性、公平性的前提下，为 DAPP 开发者打造一个完整和高效的开发生态环境，以帮助 DAPP 的快速实施和落地。

合约升级

在传统的 APP 开发领域，功能升级一直是流程里重要的一环。DAPP 的前身来自于交易，且由于“代码即法律”的区块链精神，功能升级一直没有从系统化

角度得到有力支撑。考虑到 TAS 的最终目标是构建强大的商用 DAPP 平台，我们将建立一套完整的合约升级方案，从平台层面让合约构建者和使用者可以协调功能升级方案。合约创建者通过发起合约升级邀约，合约使用者在得到邀约通知后，可以全面评估升级方案对自身利益的影响，并自由选择是否升级到新合约。

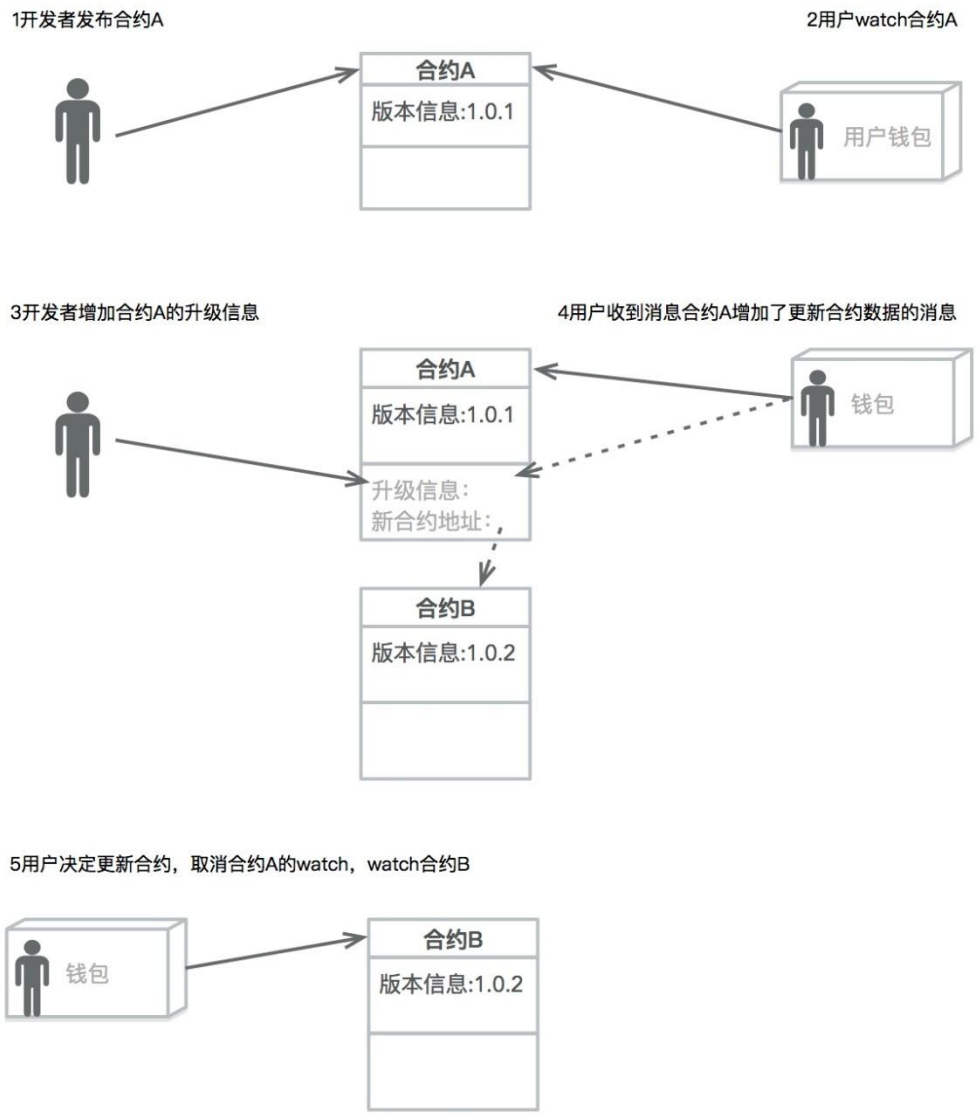

图 15. 智能合约升级

重大异常修复（硬分叉）

在传统的 APP 开发过程中，除了 APP 的日常功能性升级和 bug 修复升级外，

还有重大 bug 触发的应用强制升级和数据回滚。在以智能合约为基础的 DAPP 迭代过程中，不能单方面由研发人员决定是否进行强制升级和数据回滚，平台层面的支持有助于快速修复和解决 DAPP 运行过程中的严重问题，做到最小粒度的硬分叉。

TAS 将提供一套完善的接口和工具链，在 DAPP 发现严重问题时可以让开发者快速便捷的对运行中的 DAPP 进行分叉处理，包括合约的镜像拷贝、数据拷贝和指定高度回滚，并通知所有 DAPP 的参与者。

高效 VM

目前大部分支持智能合约的区块链系统，都采用栈式架构的虚拟机，栈式架构很好的解决了不同物理操作系统间的一致性问题，但同时也导致了运行效率低下。TAS 的 TVM 基于 LLVM 路线，在保证一致性和扩展性的基础上，显著地提高了合约的运行效率。

应用组件库

TVM 推出了一系列基础组件库，在提供高解耦性和可复用的基础上，极大的降低了 DAPP 的上链存储成本。同时，TVM 的开放架构鼓励开发者开发三方组件库，在通过严格的审计和上链后，这些三方库会向 DAPP 开发人员开放，进一步降低 DAPP 开发人员的研发成本和存储成本。随着三方库的广泛使用，开发者会收到代币激励以鼓励他们优化和创造更多的高可用和可信赖的三方库。我们认为这样的基础组件库和应用组件库模式有利于构建良好的 TAS 生态联盟。

多合约协同

对于一个大型 DAPP 来说，需要根据不同功能和分层设计等理念把代码拆分成多个水平和垂直的子智能合约，协作完成复杂的商业逻辑。考虑到合约本身的契约特性带来的不可修改性，一个设计良好的 DAPP 需要在实现过程中使用多合约协同工作，以便在功能升级时，可以用最小的代价替换原有合约中有变动的部分。我们认为一个完整的多合约之间的通信、依赖关系和打包方案，是对 DAPP 开发者必不可少的支持。

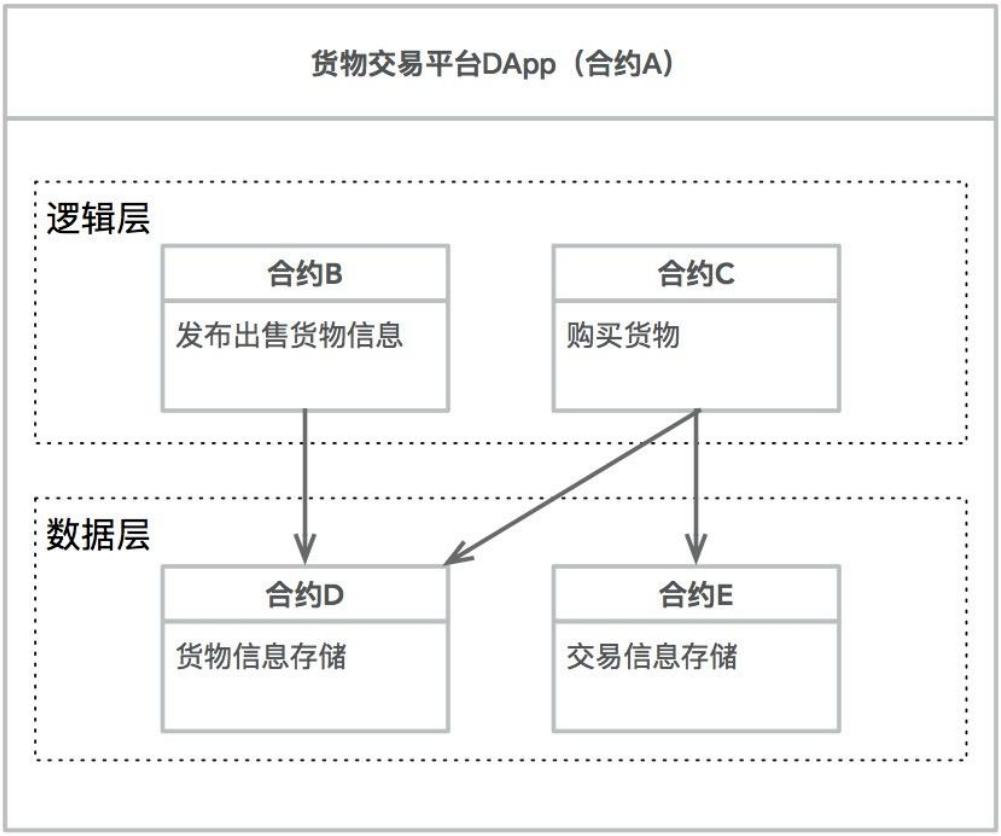

图 16. 多合约协同

自组织治理

硬分叉一直是区块链发展中面临的巨大难题和挑战。对于设计目标是支撑大规模商用 DAPP 的公链来说，如何实现自组织治理，避免硬分叉对 DAPP 带来的影响，是区块链系统设计的关键因素。TAS 尝试使用去中心化的智能协商协议 SNGP(Smart Negotiation Governance Protocol)来解决这个问题。

SNGP 通过智能合约实现了一套可编程的社区治理协议，基于这套协议不仅可以进行系统和 DAPP 的属性调整，还可以进行系统行为的升级，从而实现自组织的自我更新和迭代，尽可能避免随意的硬分叉对商用 DAPP 带来的负面影响。

在实现上, SNGP 被设计为一种投票协议，即满足一定条件的自组织成员可以发起提议，自组织成员通过设备信用或保证金质押进行 DPOS 选举投票，在得票率超过阈值后，整个自组织系统进行自动调节或升级。同时 SNGP 会根据最终结果对用户的投票行为进行奖惩和激励, 以保证用户持续的在正确的路线上进行委托或投票。在更长远的计划里，我们会支持 HTTP 和 RPC 两种通讯方式从外部引入结果，以及公投宣判人角色，在一定程度上解决链上和链下的闭环问题。

SNGP 作为一种投票协议，接受的输入是一个提议或预测，输出是该提议的投票结果或最终的自然结果。 SNGP 的作用不仅限于自组织的治理，实际上,所有可以用投票处理的场景，如预测博弈、竞选博弈等，都可以用 SNGP 公平公正的得到最优的结果。

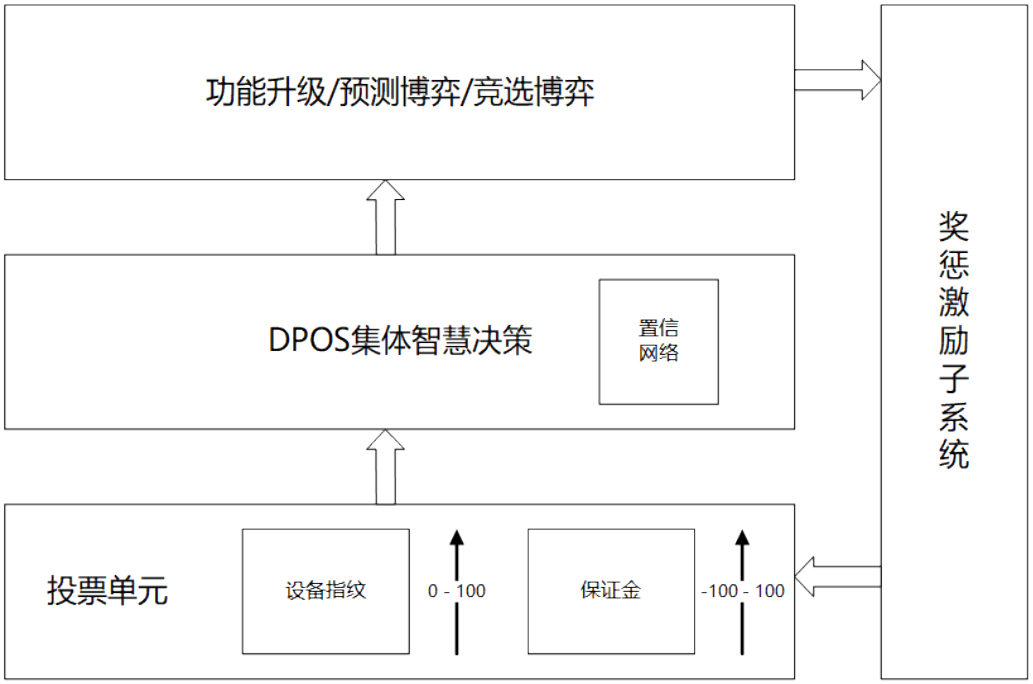
功能升级/预测博弈/竞选博弈
DPOS集体智慧决策
置信
网络
投票单元
设备指纹
0 - 100
保证金
-100 - 100
奖惩激励子系统

图 17. 自组织治理流程

代币经济模型

基于 TAS 链发行 TAS 代币，能够有效的促进 TAS 链之上数字版权、供应链金融、游戏、电子票据等商用 DAPP 的产品孵化、产品易用性和生态推广。健康的代币经济模型有助于将开发者、用户、投资者与社区建设成为一个可持续的平台，TAS 团队承诺将长期发展 TAS 公链技术。

TAS 团队在新加坡注册成立 TAS 基金会，该基金会是服务于 TAS 开源社区的非营利组织，TAS 基金会负责为 TAS 社区筹集运营资金，对参与并推动 TAS 开源项目的企业、组织或个人提供支持。该基金会作为治理主体，全面负责管理 TAS 技术开发和应用开发，维护 TAS 持有人权益，宣传推广 TAS 品牌。

TAS 代币发行量为 100 亿个，分配及使用情况如下：

（1） 私募 40%，对象为天使投资人、机构主要为 TAS 的早期发展提供资源和技术支持，有锁定。

（2） 公募 10%，对象为高净值用户，无锁定。

（3） 团队预留 18%，用于创始团队激励，有锁定。

（4） 基金会社区奖励 20%，主要用于对社区中重要参与者和贡献者奖励，激励与支持社区参与者开发各类商业应用，定期举办社区开发者活动，推进 TAS 生态发展，技术代码安全审计、法律财务等合规审查、第三方审计等。

（5） 市场合作 12%，包括：市场品牌推广、持有人权益维护、交易所等，无锁定。

所有的锁定时间都是以 ERC20 智能合约递交开始计算。

每年固定增发代币初始发行总量的 3.3%，用于补充流动性。

路线图

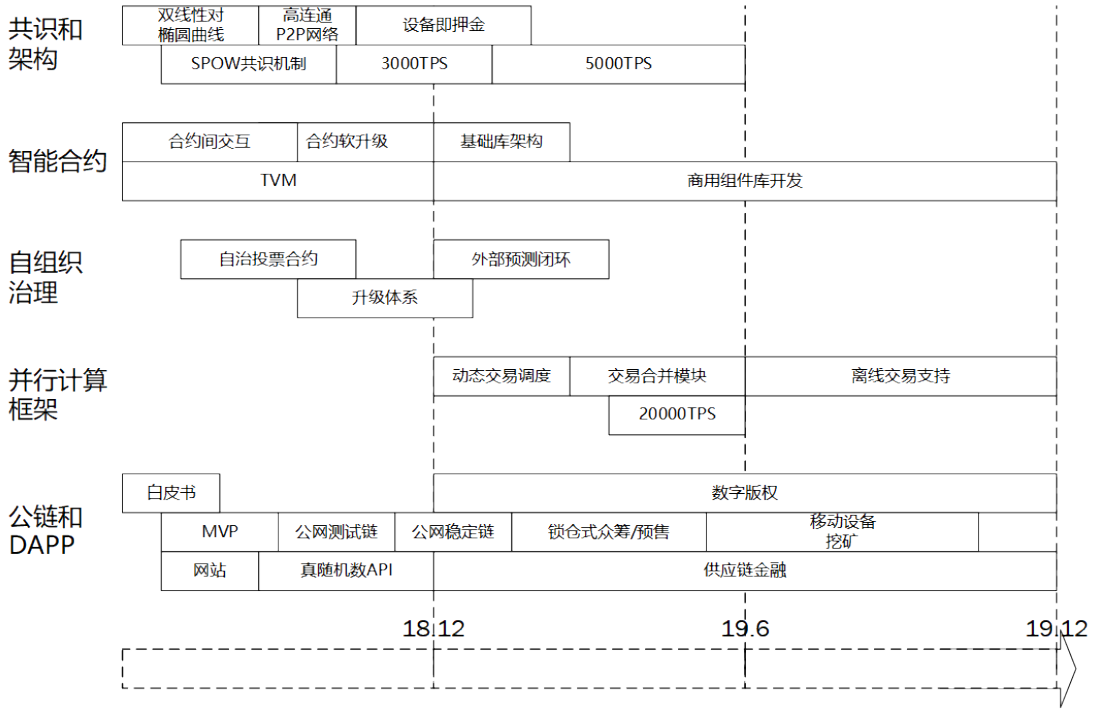

图 18. TAS 实施路线和重要时间节点

团队成员

创始团队

吴轶群

TAS 创始人&CEO。前虾米音乐 CTO&联合创始人，前阿里巴巴安全部大数据产品负责人&大数据总架构师。在 P2P、搜索引擎、个性化推荐、大数据安全等高性能分布式系统和机器学习领域上有 20 年带领团队经验。虾米音乐 DAU 最高 700 万，个性化推荐长期雄踞中国第一，阿里安全部负责双十一峰值每秒 100 万+高并发应用。

沈燕

TAS 联合创始人&COO。清华大学硕士毕业，师从长江学者毕军教授与吴建平院士，密码学和算法专家、网络安全与点对点网络协议设计专家。参与多份 RFC 的设计与编写，与 2 位导师共同的发明专利 200710178491，被广泛应用于 IPv6 网络的各种基础协议。技术极客，11 年互联网技术经验，曾任浙江融雪网络 CTO、杭州曲奇科技 CTO、网易严选算法负责人等职，在架构设计、密码学及算法领域有丰富的经验。区块链技术爱好者及投资专家，机器学习量化交易专家，对区块链技术的演变、技术社区的运营与发展有深入的研究。

吕晟珉

TAS 联合创始人&共识负责人。浙江大学应用数学系博士毕业。前淘淘搜算法组负责人，前趣搭网络研发部负责人，在 CAD&CG、图像处理、机器学习、密码学方面有 15 年以上算法研发经验。发表 2 篇美国发明专利，9 篇国内发明专利。

戴佳

TAS 联合创始&架构负责人。东南大学硕士毕业，计算机科班出身，前阿里巴巴技术专家。曾就职于华为、阿里巴巴，近十年互联网开发经验，擅长高并发场景的大规模分布式服务，包括性能优化与高稳定性。在阿里巴巴期间带领设备指纹团队，设备指纹是阿里安全最重要的基础服务之一，其数据与产品广泛用于反作弊、反欺诈、盗号等场景。带领团队对 PB 级大数据进行分析和算法实现，并提供 SaaS 服务。SaaS 服务历经多年双十一考验，峰值访问量每秒超过 40 万次。

顾问团队

胡红钢

中国科技大学信息安全系主任，研究方向密码学和信息安全。2000 年毕业于中国科学技术大学数学系，同时获得电子工程与信息科学系双学位。随后分流至中国科学院信息安全国家重点实验室，2005 年获得博士学位。2005 年至 2007 年在中国科学院软件研究所工作。2007 年至 2011 年在加拿大滑铁卢大学应用密码学研究中心工作，随后加盟中国科学技术大学。

战略合作

技术是商业的基础而不是全部，TAS 通过解决信任问题而大幅降低信用流通中间环节的成本。我们认为 DAPP 最有可能在供应链金融、数字版权、电子票据、游戏等领域率先爆发，只有 DAPP 的全面落地才代表着区块链时代的真正到来。同时随着区块链全面覆盖互联网，以及独特的去中心化特性，区块链安全的重要性日益凸显。TAS 会在供应链金融、电子票券、数字版权、游戏、众筹和预售等多个领域和商业伙伴进行合作，我们也希望有更多志同道合的商业伙伴一起参与到 TAS 大生态的建设。

参考文献

[1]. Satoshi Nakamoto, “Bitcoin: A peer-to-peer electronic cash system, In: URL http://bitcoin.org/bitcoin.pdf,” 2008.

[2]. Vitalik Buterin. “Ethereum: A next-generation smart contract and decentralized application platform.” In: URL https://github.com/ethereum/wiki/wiki/White-Paper , 2014.

[3]. NIST, “Sha-3 standard: Permutation-based hash and extendable-output functions,” 2015.

[4]. F. Vercauteren, Optimal pairings, IEEE Transactions on Information Theory, vol. 56, no. 1, pp. 455–461, 2010.

[5]. l lvm. https://llvm.org/. Accessed: 2017-08-01.

[6]. Bitcoin Computation Waste, http://gizmodo.com/the-worlds-most-powerful-computer-network-is-being-was-50 2013.

[7]. S. Micali. Algorand: The Efficient Public Ledger. https://arxiv.org/abs/1607.01341.

[8]. S. Micali, M. Rabin and S. Vadhan. Verifiable Random Functions. 40th Foundations of Computer Science (FOCS), New York, Oct 1999.

[9]. Y. Gilad, R. Hemo, S. Micali, G. Vlachos, and N. Zeldovich. Algorand: Scaling Byzantine Agreements for Cryptocurrencies.

[10]. DFINITY White Paper: Consensus System. In URL https://dfinity.org/pdf-viewer/pdfs/viewer?file=../library/dfinity-consensus.pdf. 2017.

[11]. B. David, P. Gaˇzi, A. Kiayias, A. Russell, Ouroboros Praos: An adaptively-secure, semi-synchronous proof-of-stake blockchain. EUROCRYPT 2018.

[12]. Kademlia: A Peer-to-peer Information System Based on the XOR Metric, In URL https://pdos.csail.mit.edu/~petar/papers/maymounkov-kademlia-lncs.pdf

[13]. STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs). In URL https://datatracker.ietf.org/doc/rfc3489/

[14]. Sharding FAQs In URL https://github.com/ethereum/wiki/wiki/Sharding-FAQs

[15]. Jeffrey Dean and Sanjay Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. OSDI 2004

[16]. wanghao, "Digital product selling and sharing method based on point-to point document exchange system", In: URL https://patents.google.com/patent/CN1889119A/en, 2006

[17]. Miguel Castro and Barbara Liskov. Practical byzantine fault tolerance. In OSDI, 1999.

[18]. https://lightning.network/

[19]. https://raiden.network/.

[20]. Shamir A. How to share a secret[J]. Communication of the ACM, 1979,22(11):612-613.