Polkadot 概述及其设计方案

Jeff Burdges[1], Alfonso Cevallos[1], Peter Czaban[1]
Rob Habermeier[2], Syed Hosseini[1], Fabio Lama[1],
Handan Kılınc Alper[1], Ximin Luo[1], Fatemeh Shirazi[1],
Alistair Stewart[1], Gavin Wood [1,2]

[1] Web3 Foundation,
[2] Parity Technologies

June 1, 2020

译者序

本文是波卡技术论文，包括核心技术设计：混合共识 GRANDPA/BABE、中继链状态机、经济模型、治理模式、XCMP/SPREE 等内容。本文对于全面深入理解 Polkadot 具有较大的价值，适合具有一定区块链基础的读者阅读。针对波卡技术的设计思路，可进一步参考 Polkadot Wiki；如果想要更深入了解设计背后的代码逻辑，可参考 Polkadot github 进一步研究。

本文第 1 章、第 2.1 节、第 4.3.2-4.6.2 节由郭斌翻译；第 2.2-2.3 节、第 4.6.3-4.8.7 节、第 5 章、附录 A.3、词汇表 B 由王飞凤翻译；第 3 章、第 4.1-4.3.1 节、附录 A.1-A.2 由冯路翻译；郭斌负责全文统稿及图片重制。

最后，感谢 Web3 基金会、Parity 亚洲团队的支持，特别感谢 Jimmy Chu 对具体翻译工作：专业名词解释、专业内容表达的建议和支持，感谢 Helena、Qinwen Wang 对本文翻译工作筹建的建议和支持。

概 要

在本文中，我们描述了异构多链协议 Polkadot 的设计组件，并解释了这些组件如何帮助 Polkadot 解决区块链技术的一些现有缺点。目前，现有的很多不同功能的区块链项目，可能在设计时未考虑相互协作的能力。这导致用户难以在不同的区块链之间使用大量交互的应用程序。此外，随着项目数量的增加，每个项目单独提供的安全性变得越来越弱。Polkadot 旨在为多条链提供一个可扩展且可互操作的框架，该框架具有池化安全性，这是通过本文中描述的组件集合实现的。

Translator's Preface

This paper is a technical paper on Polkadot, including the core technical design: hybrid consensus GRANDPA/BABE, relay chain state machine, economic model, governance model, XCMP/SPREE, etc. This paper is of great value for a comprehensive and in-depth understanding of Polkadot, and is suitable for readers with certain blockchain foundation. For the design ideas of Polkadot technology, you can further refer to Polkadot Wiki; if you want to understand the code logic behind the design more deeply, you can refer to Polkadot github for further research.

Chapter 1, Section 2.1, and Section 4.3.2-4.6.2 are translated by Bin Guo; Sections 2.2-2.3, 4.6.3-4.8.7, Chapter 5, Appendix A.3, and Glossary B are translated by Feifeng Wang; Chapter 3, Sections 4.1-4.3.1, and Appendix A.1-A.2 are translated by Lu Feng; Bin Guo is responsible for the full-text unification and image reproduction.

Finally, we would like to thank Web3 Foundation and Parity Asia team for their support, especially Jimmy Chu for his suggestions and support on specific translation work: explanation of professional terms and expression of professional contents, and Helena and Qinwen Wang for their suggestions and support on the preparation of translation work of this paper.

Abstract

In this paper we describe the design components of the heterogenous multi-chain protocol Polkadot and explain how these components help Polkadot address some of the existing shortcomings of blockchain technologies. At present, a vast number of blockchain projects have been introduced and employed with various features that are not necessarily designed to work with each other. This makes it difficult for users to utilise a large number of applications on different blockchain projects. Moreover, with the increase in number of projects the security that each one is providing individually becomes weaker. Polkadot aims to provide a scalable and interoperable framework for multiple chains with pooled security that is achieved by thecollection of components described in this paper.

目　录

1.简介

Internet 最初是为 TCP/IP 等去中心化协议设计和构建的，但是，它的商业化导致了当今所有流行的 Web 应用程序的中心化。我们指的不是物理基础设施的任何中心化，而是指逻辑中心化对基础设施的权力和控制。两个突出的例子是像谷歌和 Facebook 这样的大公司：虽然它们以物理上分散的方式在世界各地维护服务器，但这些最终都由一个实体控制。

控制系统的中央实体会给每个人带来许多风险。例如，他们可以随时停止服务，可以将用户的数据出售给第三方，并在未经用户同意的情况下操纵服务的运作方式。这对于严重依赖这些服务用于商业或私人目的的用户尤其重要。

随着个人数据所有权意识的觉醒，网络用户对更好的安全性、自由度和控制的需求日益增长。因此，对于没有单一实体控制系统的更分散的应用程序而言，这是一种反向运动。这种权力下放的趋势并不新鲜。它已被用于网络和其他系统开发的许多领域，例如自由软件运动。

区块链是为解决这些问题而提出的一项技术，旨在建立一个去中心化的网络。然而，它只有在大众可用的情况下才能与集中式网络竞争的最终用户。其中一个重要方面是，单独的应用程序必须能够交互，否则每个应用程序都会变得孤立，不会被尽可能多的用户采用。必须建立这样一个互操作性机制引入了新的挑战，由于两种范式之间信任模型的根本差异，集中式模型中缺少许多挑战。例如，比特币[14]和以太坊[10]是工作量证明(PoW)区块链，其中安全性依赖于对处理能力的假设；而股权权益证明(PoS)系统的安全性依赖于激励措施和销毁保证金的能力。这些差异给区块链之间的互信带来困难。区块链技术需要解决的另一个挑战是可扩展性。现有的区块链系统普遍存在高延迟，每秒只能进行数十笔交易[14]，而 Mastercard 或 Visa 等信用卡公司则执行每秒数千笔交易[3]。

区块链可扩展性的一个突出解决方案是并行运行许多链，通常称为分片。Polkadot 是一个多链系统，旨在将所有这些链的安全力量集中在一个共享的安全系统中。它于 2016 年由 Gavin Wood[33]首次引入，在本文中我们将详细介绍。

简而言之：Polkadot 利用称为中继链的中央链与称为平行链（平行链的组合）的多个异构和独立分片链进行通信。中继链负责为所有平行链提供共享安全，以及平行链之间的可信跨链交易。换句话说，Polkadot 旨在解决如上讨论的内容：互操作性、可扩展性以及因算力分流所削弱的安全性问题。

论文组织：在下一节中，我们将概述 Polkadot 网络，包括其与客户端平行链的外部接口，我们将在后续部分中对其进行扩展。我们回顾初步信息，例如第 3 节中对 Polkadot 参与者角色和我们的对手模型的描述。我们在第 4 节中解释了 Polkadot 的子协议和组件试图实现的目标，然后继续详细阐述它们，包括底层加密和网络原语。最后，我们将在第 5 节中简要讨论一些未来的工作。在附录中，我们回顾了相关工作，例如与其他多链系统 A.3 的比较、桥接到外部链 A.2 的互操作性方案的简短描述、用于消息传递的安全执行方案 A.1，以及包含 Polkadot 特定术语的词汇表 B。

2.概要

本节目的是描述 Polkadot 的主要功能，而不会详细介绍设计方案和推论。

Polkadot 系统由一个称为中继链的开放协作去中心化网络构成，该网络与许多其他并行运行的外部链交互，称为平行链。从上层的角度来看，平行链是中继链的客户端，中继链为这些客户端提供安全服务，包括安全通信。这是中继链的唯一目的；平行链是提供应用程序级功能的载体，例如加密货币。

平行链的内部细节不是中继链关心的问题；平行链只需要在我们指定的接口上显示。其中一些期望是基于区块链的基础组成部分，因而得名。但是，其他非区块链系统也可以作为 Polkadot 平行链运行，只要它们满足该接口即可。如下所述：相关部分加下划线。

综上所述，Polkadot 是一个可扩展的异构多链协议。

2.1 安全模型

我们假设平行链作为中继链的外部不可信客户端运行，并且中继链仅通过接口处理平行链，而不对其内部进行假设。例如，在内部它们可能为许可或开放网络；如果一些内部用户破坏平行链内部结构，从 Polkadot 的角度来看，该平行链（作为单个客户端载体）都是恶意的。

作为开放的去中心化网络，Polkadot 中继链需要内部处理某种程度的恶意行为。特定的单个节点是不可信的，但数量有限且随机组成的节点集合是可信的，该协议的作用是确保外部的中继链作为一个整体是可信的。有关详细信息，请参阅第 3.2 节。

2.2 节点与角色

Polkadot 中继链网络由节点和角色组成。节点是物理执行 Polkadot 软件的网络级实体，角色（第 3.1 节）是执行特定目的的协议级实体。节点可以扮演多种角色。

在网络层面，中继链是开放的。任何节点都可以运行软件并作为以下任何类型的节点参与：

1. 轻客户端 - 从网络中检索某些与用户相关的数据。轻客户端的可用性无关紧要 - 它们不会对其他节点/客户端提供服务。

2. 全节点 - 检索所有类型的数据，长期存储，并与其他全节点同步通信。因而必须是高可用的。

   (a) 哨兵节点 - 公共可访问的完整节点，为私有完整节点执行受信任的代理服务，由同一运营商运行。

   有时我们指的是平行链的完整节点。对于由非区块链所构成的平行链来说，这意味着他们参与到足够的程度，以至于他们可以验证通过它的所有数据。

除了分发数据之外，中继链节点还可以执行下面列出的某些协议级别的角色。其中一些角色具有与之相关的限制和条件：

1. 验证人[1] - 执行大部分安全工作。必须是中继链的全节点。与平行链收集人交互，但不需要作为全节点参与平行链出块工作。

2. 提名人 - 支持和选择验证人列表的利益相关者（第 4.1 节）。可以由轻客户端完成，它们不需要对平行链有任何了解。

平行链可以决定自己的内部网络结构，但预期通过如下角色与 Polkadot 交互：

1. 收集人 - 收集平行链数据并将其提交给中继链，遵守以下描述的协议规则。它们是由平行链定义选择的，并且必须是完整节点。

2. 钓鱼人 - 代表提供奖励的中继链对平行链的正确操作进行额外的安全检查。这个角色是自我分配和奖励激励的，并且必须是平行链的完整节点。

2.3 协议

波卡中继链协议，包括与平行链间的交互，其工作原理如下。

1.对于平行链:

(a) 收集人会实时跟踪中继链区块的生成过程和共识协议，分别执行下面的步骤(2)和(5)。例如，作为全节点参与到中继链当中，基于此来确定最有可能成为最新中继链的区块。另一方面，最新平行链区块（或其他数据）也将由这一最新中继区块所确定。

(b) 收集人对上述最新平行链区块上构建的数据完成签名后，将信息以间接形式递交到其平行链委派的验证人(平行链验证人简称验证人)，通过此步骤将信息输送到中继链。理想情况下，为提高执行性能，收集人仅递交唯一的方案。

(c) 由平行链验证人决定支持哪一个平行链区块，并公布该区块的相关数据，以表明其将作为该平行链的候选人被添加至下一个中继区块当中。

2.中继链上负责区块生成的验证人会从所有平行链上收集候选区块，并把这些候选区块和最新的中继链外部调用一起放入中继链最新生成的区块中(第 4.3.1 节) 。考虑到执行性能，这一过程产生的数据不包含平行链的完整数据，仅包含元数据和部分数据，当然安全相关的元数据包含在内。

在不利的情况下，这可能导致分叉，步骤(5)中会给出详细说明。该子协议被设计成即使有分叉，参与者也能知道最有可能成为最终块的区块，类似于工作量证明协议。

3.子协议的运行以确保完整数据确实可用、涵盖并分发到其他各种中继链节点。（第 4.4.2 节）

4.平行链递交数据时可能包含相关其向另一条平行链发送信息的相关信号指示，包括促进该过程的元数据。现在这些数据将包含在中继链头部分，所以作为接收方的平行链可以得到新信息输入的相关信号。相比当前，接收方需要通过检索发送方的信息正文才能获取相关信息。（第 4.4.3 节）

1 验证人在某种意义上也是中继链网络的收集人，由于验证人也进行中继链网络下的外部信息收集（例如交易信息）。然而，虽然验证人也会执行这类任务，但我们还是将其定义为“验证人”。同时，术语“收集人”仅针对平行链收集人。

5.验证人提交他们对区块的投票并最终确定，解决了因意见不同而产生分叉的问题（章节 4.3.2）。上述投票将会被添加到中继区块中。
本文接下来的内容将对上述内容进行阐述-第 3 节对角色设计进行详述，第 4 节对协议子组件进行详解。

3 序言

在本节中，我们将更详细地介绍参与波卡运行的不同角色，包括我们为这些角色而创建的安全模型。对于这些角色的理解，有利于进一步理解整个协议的设计，包括协议是如何工作的，以及采用这种设计的原因。

3.1 角色定义

波卡网络节点的运行是基于接下来我们要介绍的相关角色和功能设定而展开的。

验证人：验证人在波卡网络拥有最高权限，控制并承担新区块的打包工作。验证人需要质押足够多的资金，但是由于我们允许其他有资金的提名人推举一个或多个可以代表他们的验证人，所以验证人的部分资金可能并不是自有资金，而是来自提名人。各验证人运行中继链的客户端，必须具备高可用性和高带宽的性能条件，节点必须准备好在每一个中继区块上批准某一平行链的新区块，有时候可能是几个新区块的确认。这个过程包括接受、验证、再发布候选区块。平行链的任务分配给验证人存在随机性，且变化频繁，要求验证人维护所有平行链的数据并保证数据库的完全同步，显然是不合理的，于是，“收集人”概念应运而生，作为第三方平行链的新区块。指定验证人集合一旦合理地批准所属平行链的所有新区块，验证人本身必须进行中继链的区块批准工作。过程包括更新交易队列的状态（将数据从一条平行链的输出队列传输到另一条平行链的输入队列），处理已批准的中继链的交易批次以及对最终区块进行审批。如果查证到验证人没有达到职责要求，将会被重罚。例如，质押在他们名下的所有资金或者部分资金被没收。某种意义上，验证人角色和基于 POW 区块链的矿池有类似之处。

提名人：提名人作为利益相关方，为每一个验证人贡献安全性资金。提名人主要作用就是将风险资本质押到他们信任的一个或一组验证人，以代表他们行使维护网络的职责，除此之外，无其他角色职责安排。根据其资金贡献量，提名人会得到其资金占验证人总质押金额里相应比例的奖励或惩罚。和收集人一样，某种意义上，提名人和基于 POW 网络的矿工相类似。

收集人：交易收集人（简称收集人）作为帮助验证人生产有效平行链区块的一方，会运行某个特定平行链的全节点，也就是说收集人需要保留所有授权新区块所必需的信息，用于打包新块并执行交易，就跟基于 PoW 区块链的矿工一样。在正常情况下，收集人收集并执行交易，并创建一个“未封装”的区块，连同有效性证明信息将区块递交给一个或多个当前负责审查该平行链区块的验证人。

钓鱼人：不像其他的两个参与方，钓鱼人并不直接参与区块打包的过程。他们是独立的"赏金猎人"，激励他们的是一次性的大额奖励。严格来说，我们希望通过“钓鱼人”的设置，减少恶意行为的发生。即使发生类似情况，希望也只是因为资金质押方私钥不小心泄露，而不是出于蓄意的恶意企图。该名字缘由是考虑到其期望奖励的频率，选择参与的最小要求以及最终能够获得奖励的数量。钓鱼人的奖励来自于能够及时发现资金质押方的非法行为，这对于检测无效平行链区块的批准是非常有价值的。钓鱼人通过及时证明至少有一方（收集人或验证人）存在作弊行为而获得奖励，这对监控无效平行链区块生成与批准很有价值。

钓鱼人有点类似于区块链系统中的全节点，所需的资源相对较少，并且不需要承诺稳定的正常运行时间和带宽。他们的不同之处在于钓鱼人必须绑定一小笔保证金。这部分绑定的保证金可以防止因女巫攻击而浪费验证人的时间和计算资源。虽然钓鱼人是波卡安全模型的一部分，但由于并没有为钓鱼人设计激励模型，所以对 Polakdot 的设计需要保证在没有他们的情况下系统也很安全。我们未来工作的一部分既是为钓鱼人增加一个激励模式。

如图 1，图例展示了 Polkadot 协议中定义的结构元素和不同的角色：拥有 6 条平行链、18 个验证人以及每条平行链拥有 5 个收集人。图 2 显示了包含 5 个中继区块的中继链。需要注意的是，分配给一条平行链的验证人数量，是通过验证人总数除以平行链的数量来决定的，但收集人的数量是独立于平行链数量的。桥是允许外部链与 Polkadot 互操作的子协议，更多信息请参见 A.2。

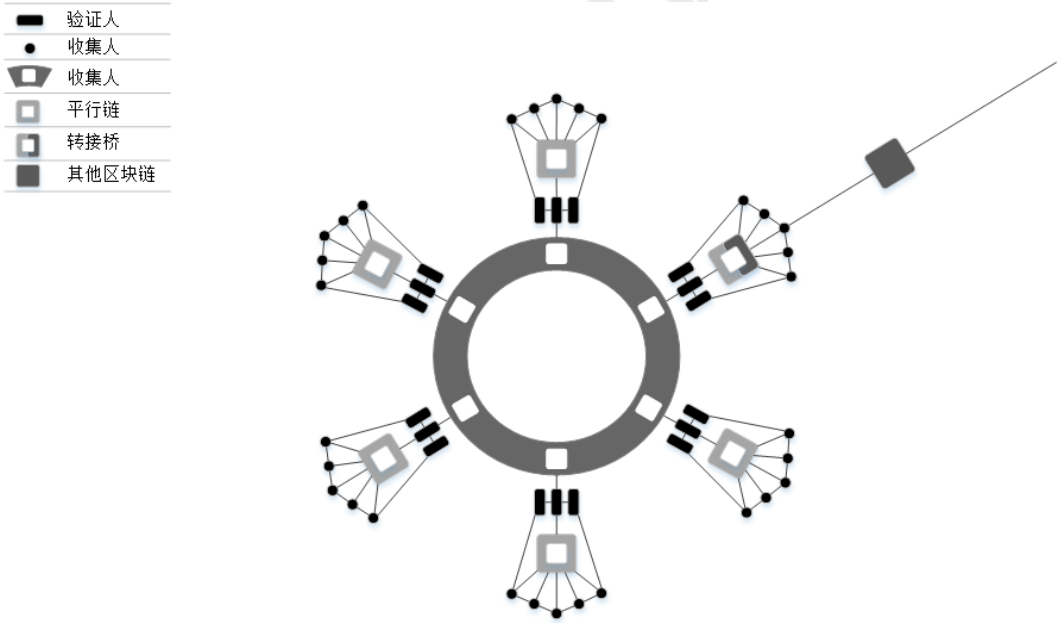

图 1：一个保护六个平行链区块的的中继块。每个平行链有 5 个收集人和 3 个验证人

3.2 Polkadot 对抗模型

角色：通常我们假设诚实的一方会遵循协议算法，而恶意的一方可能采用任意的算法。我们假设四分之三的提名人质押行为属于诚实的一方。基于这一假设，被提名人选出的验证人有三分之二以上是诚实的。我们对恶意钓鱼人的数量没有设定任何限制，因为他们的恶意行为可以被发现并受到惩罚。

平行链：我们对平行链的区块生成机制没有任何安全假设。另一方面，我们假设大量的收集人是诚实的。Polkadot 的安全性并不取决于任何特定的诚实收集人，但它需要存在一些诚实的收集人。

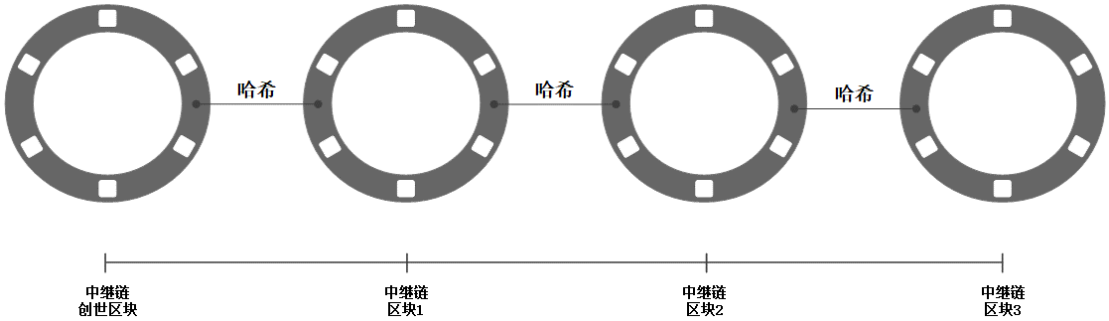

图 2:拥有五个中继区块的中继链。其中每个中继区块包含来自六条平行链的区块，但每个中继区块包含的平行链区块数量可能是不一样的

部分协议假设每条平行链至少有一个诚实成员；如果这个不可行或不现实，我们就不遵循这一假设，而是对完全恶意的成员进行额外检查。

密钥：我们假设恶意方通过任意算法生成密钥，而诚实方总是安全地生成密钥。

网络和通信：所有验证人都有他们自己的本地时钟，且不依赖于任何中央时钟。我们假设验证人和收集人处于部分同步的网络中，意味着，在一个未知参数之下，在大多数时间单位Δ（一个未知参数）之后，验证人或收集人发送的消息将会到达网络中的所有各方。所以我们假设信息的最终传递是在 Polkadot 上。我们还假设收集人和钓鱼人可以连接到中继链网络中来提交他们的报告。

4. 组件和子协议

接下来，我们简洁而全面地总结波卡的功能性，以及继续阐述单独的组件和子协议。

波卡的验证人采用 NPoS 选择机制（第 4.1 节）。NPoS（Nominated Proof-of-Stake）是基于对 PoS 的改进，它允许任意数量的 Token 持有者以提名人的身份参与到网络中，他们的质押支持了大量但有限的验证人集合。这个范式同时实现了高安全性和可扩展性，这个范式可同时实现高安全性和可扩展性，并通过投票机制中为人所共知的比例代表制[29][7]，来达到一个前所未有的去中心化水平。提名人在经济上赋予系统安全性，对验证人的表现起着监督者的作用。基于提名人对候选人表达的偏好，每一个 era 系统都会选择一组获得的质押数量支持尽可能高的、提名人分布尽可能均匀的验证人集合。同时，提名人也可能会因为把自己的选票贡献给了太少数量的验证人而在经济上不受激励，这有助于随着时间的推移持续保持系统的分散度。此外，选举机制对突然的变化具有很强的适应性，比如一些验证人在被大幅削减后被踢出，选举机制此时会自动在一组新的验证人之间重新分配提名人的支持，即使选票本身没有变化。

波卡的安全性目标是参与者在理性的的情况下实现拜占庭容错（见第 4.5 节的关于激励和经济的更多细节）。我们认为在 NPoS 机制下，质押人选择的一组验证人集合中，至少有 2/3 的验证人是诚实的。

被选中的验证人集合负责运行中继链（第 4.2 节）。每个平行链的收集人负责生产平行链区块（第 4.4.1 节），验证人被分为轮动的子集，一个平行链一个子集，在这些平行链区块的区块头被纳入中继链之前需要被验证有效性。

为了实现良好的可扩展性，每个子集中的验证人数量都很小。尽管如此，NPoS 机制保证每个验证人节点获得良好支持的，可用性和有效性机制（第 4.4.2 节）可以确保任何针对波卡有效性的攻击都是可预期且非常昂贵的。实际上，波卡的整体经济安全性会支持到每一条平行链。这和最初的想法形成鲜明对比：比如使得 100 条独立的平行链拥有价值相等的质押数量，即平均每条平行链被 1/100 的总质押支持，所以每条平行链只能从 1/100 的安全性中获益。我们通过对每个平行链进行擦除编码来保证可用性，使验证人对这些平行链区块的可用性负有共同责任，而不会破坏其可扩展性。

为了使其发挥作用，我们需要能够平行链具备回滚功能，直到我们大概率确定所有平行链都是正确的。这意味着，我们需要具备重构链的能力，为此，链需要能够进行分叉。因此我们采取了 BABE（第.3.1 节）作为块生成机制，虽然由验证人运行，但具有类似 PoW 链的特性。具体来说，我们采用最长链规则作为我们共识的一部分，所以并不会预先知道谁是下一个区块生产者。就其本身而言，BABE 要求我们从一个区块产生的那一刻起，到它被最终确定的那一刻，即当我们可以确信该区块永远不会被回滚时，需要等待很长的时间。在某些情况下，为了确保区块的可用性，需要使终态化进程放缓。但大多数时候，我们更倾向于快速确认区块终态。为此，验证人使用 GRANDPA（第 4.3 节）来确定区块的终态，这是一个与区块生成完全分离的终态敲定工具。BABE 和 GRANDPA 的分离运行，使得 GRANDPA 具有自动调节性，并允许我们延迟确认区块终态直到其可用性被验证，但这并不会减缓区块的生成速度。GRANDPA 在确定一个区块终态时使用拜占庭协议，并允许我们向跟踪验证人集合的实体证明哪些块已经被确定终态，这对转接桥来说很重要（见附录 A.2）。

如果一个平行链上的账户给另一个平行链上的账户发送 Token，那么 XCMP（第 4.4.3 节）保证这条信息被准确传递。它发送的速度不依赖于区块被终态化需要的时间，这意味着信息发送过程中需要处理波卡可能分叉的情况。因此，我们会乐观执行，基于假定平行链区块都是正确的。如果有一个区块不正确，就需要回滚它，同时，重要的是平行链只接收来自于新的中继链分叉上的区块发送信息，而不是接收被回滚了的分支上的信息。因此，我们需要平行链和 XCMP 逻辑来确保中继链的一个分叉定义了一个一致的 Polkadot 历史数据，因此消息只有在这个中继链分叉定义的历史中被发送时才会到达。

如果 Token 的交易结合 SPREE 模块（附录 A.1) 来进行，则可以保证只要平行链正确运行，Token 只能以约定的方式被创建和销毁，同时可用性和有效性机制保证了链上代码的正确执行。SPREE 也确保 Token 交易逻辑所需的共享代码是正确的。即使平行链可以改变他们自己的代码，却不能改变 SPREE 模块的代码。相反，SPREE 模块的代码是集中存储的，该代码的执行及存储将与状态转换的其他部分进行沙盒处理。这就保证了 Token 交易消息被正确解析，且确保我们获得了想要 Token 的保证。

在经济方面(第 4.5 节)，我们的目标是有一个可控制的接近恒定的年通货膨胀率。如前所述，对于系统的安全性来说，非常重要的是所有验证人都有大量的质押金支持。我们为验证人和支持他们的提名人制定了自适应的奖励计划，确保了 NPoS 的整体参与度保持在较高水平，并且验证人的质押支持是均匀分

布的。在更细颗粒度的层面上，我们会根据验证人每次的执行行为来其进行支付或削减，并按比例对提名人进行同样的奖励或惩罚，以确保合理的策略和诚实的行为相一致。

中继链自身的逻辑会不定期的升级。治理机制（第 4.6 节）允许波卡 Token 持有者参与到决策流程，而不是通过中心化的权利来对系统作出任何改变——或者是类似一些去中心化系统，通过团队开发者来决定，他们一次有争议的代码改变通常会导致区块链陷入僵局或永久分岔。我们希望有一个机制可以平衡系统，可以在需要的时候快速做出没有争议的改变，同时也可以有工具对有争议的提议做出果断且正确的回应。波卡最终的决定，对所有重要的决定比如代码改变，通过 DOT 持有者参与的质押加权的民主公投来决定如何回应。被选举出来的委员会，负责做小一点的决定，以及在一些情况下被适当的被赋予了公投的优先权，如此他们便不能阻止大多数人都想要的改变。

最后，我们回顾了 Polkadot 子协议使用的一些原语，如第 4.7 节和第 4.8 节中分别介绍的密码学和网络方案。Polkadot 需要将无需信任的单链标准的点对点通信网络扩展到多链系统，在多链系统中，任何节点的网络流量都不应随系统的总数据而膨胀。

4.1 NPoS 和验证人选举

波卡原生 Token 是 DOT。出块采用自定义的提名人权益证明 NPoS，是具有确定性终态的共识协议，例如在 Polkadot 中，需要一组注册的验证人集合，其规模是有数量有限的。Polkadot 将维护数据为n_{val}的验证人集合。这个数字最终由治理决定，并随着平行链数量的增加线性增长；但验证人的数量将独立于网络中的用户数量，以确保网络的可扩展性。NPoS 允许数量不限的 DOT 持有者作为提名人参与，他们通过质押来提供更多价值，以帮助网络维持高水平的安全性。因此，NPoS 不仅比工作量证明（PoW）更有效，而且比传统形式的 PoS（如 DPoS 和 BPoS）更安全。即我们引入了迄今为止任何其他基于 PoS 的区块链都无法比拟的去中心化新的保证方式。

根据提名人的参数设置，在每个 era 开始时会选出一组新的验证人来为这个 era 服务，一个 era 大约为一天的时间（见附录中的表 1）。更准确地说，任何 DOT 持有者都可以选择成为验证人候选人或提名人。每个候选人的参数都表明了它获得的质押数量支持以及它希望获得的佣金回报。反过来，每个提名人则需要锁定一些质押并公布一个包含它信任的任意数量候选人的列表。然后，下面讨论的公共协议将这些列表作为输入，并选出支持最多的候选人作为下一个 era 的验证人。

提名人通过质押 DOT 支持验证人，并获得相应他们 DOT 质押占比的奖励或罚金；更多详细信息，请参阅第 4.5 节。因此提名人在经济上被激励充当系统的监督者，他们将根据验证人的质押水平、佣金、过去的表现和安全性实践等参数来确定他们将选择验证人。我们的方案允许系统选举拥有大量股权的验证人——DOT 持有量远高于任何一方的——从而有助于将验证人选举过程变成精英选举而不是富豪统治。事实上，在任何给定时刻，我们希望所有的 DOT 供应中有相当大一部分都被质押在 NPoS 中。这使得对抗性实体很难让其验证人当选，因为

它需要大量的 DOT 或足够高的声誉来获得所需的提名人的支持，而且攻击成本很高，因为它可能会失去所有的股权和赢得的声誉。

Polkadot 通过去中心化协议选举验证人，该协议具有精心挑选的、简单的和公开的规则，将提名人的可信候选人名单作为输入。从形式上看，该协议解决了一个基于批准投票的多赢家选举问题，其中提名人的投票权与他们的利益成正比，目标是去中心化和安全。

去中心化：我们的去中心化目标转化为投票理论中的经典概念：比例代表。也就是说，也就是说，一个委员会应该代表选民中的每一个少数群体，与他们的总票数（在这种情况下，他们的利益）成正比，没有一个少数群体的代表人数不足。我们在此强调，提名人--以及他们所信任的候选人名单--是衡量社会大众偏好的重要标准，不同的偏好和派别自然会出现，不仅是由于经济和安全方面的原因，还有政治、地理等原因。在一个分散的社区中，这种观点的多样性是值得期待和欢迎的，重要的是让所有少数民族参与决策过程，以确保用户的满意度。

设计一种实现比例代表制的选举制度的想法在文献中已经存在了很长时间。特别值得注意的是斯堪的纳维亚数学数学家 Edvard Phragmen 和 Thorvald Thiele 在 19 世纪末的工作。最近，学术界作出了相当大的努力来正式确定比例代表制的概念，并重新审视了 Phragm´en 和 Thiele 的方法，在算法上对其进行优化。我们的验证人选择协议是对 Phragm´en 方法的改进，且确保遵守*比例合理代表制* (PJR) [29][7]的技术特性。从形式上看，如果每个提名人$n \in \mathcal{N}$，质押 $stake_n$ 并支持一组候选人$\mathcal{C}_n \subseteq \mathcal{C}$，协议就会选择一组$n_{val}$验证人[2]$\mathcal{V} \subseteq \mathcal{C}$ ，如果这里有少数提名人$\mathcal{N}' \subseteq \mathcal{N}$，那么:

$$|\cap_{n\in\mathcal{N}'} C_n| \geq \mathrm{t} \text{ and } \frac{1}{t}\sum_{n\in\mathcal{N}'} stake_n \geq \frac{1}{n_{val}}\sum_{n\in\mathcal{N}'} stake_n$$

对一些 1 < t < n_{val}，则有$|\mathcal{V}\cap(\cap_{n\in\mathcal{N}'} C_n)| \geq \mathrm{t}$。换句话说，如果少数$\mathcal{N}'$至少有 t 个普遍信任的候选人，它可以“负担得起”提供至少$\frac{1}{n_{val}}\sum_{n\in\mathcal{N}} stake_n$权益的平均支持（即，选出的集合$\mathcal{V}$中的平均验证人支持上限)，那么这个少数群体就可声称在$\mathcal{V}$中至少有 t 个候选人，尽管不一定普遍信任。

安全性：如果一个提名人得到两个或更多受信任的候选人，从而当选为验证人，共识协议还必须确定如何分配提名人的质押份额并将收益分配给他们。相对应地，这些分配说明了每个验证人接收到的总质押支持。我们的目标是使这些验证人的支持率尽可能高且平衡。特别是，我们专注于最大化最小验证人的支持。直观地说，最小支持度对应于对手获得对一个验证人的控制成本的下限，也对应于因不当行为带来的可削减金额的下限。

如果每个提名人$n \in \mathcal{N}$质押 $stake_n$，且支持候选人子集$\mathcal{C}_n \subseteq \mathcal{C}$，那么该协议不仅必须选出一组$\mathcal{V} \subseteq \mathcal{C}$拥有 PJR 属性的验证人 n_{val}，而且还需要根据提名人支持且被选中的验证人中定义一个分布，即函数$f: \mathcal{N} \times \mathcal{V} \to \mathbb{R}_{\geq 0}$

2 为便于表述，我们在这里考虑的是候选人本身没有质押的模型。一般情况下，可以通过将每个候选人的质押表示为一个专门提名该候选人的额外提名人，来简化模型。

$$\sum_{v \in \mathcal{V} \cap \mathcal{C}_n} f(n, v) = stake_n \text{ for each nominator } n \in \mathcal{N}$$

$$\max_{(\mathcal{V}, f)} \min_{v \in \mathcal{V}} support_f(v), \quad \text{where } support_f(v) \coloneqq \sum_{n \in \mathcal{N}: v \in \mathcal{C}_n} f(n, v)$$

这个目标所定义的问题在文献[30]中被称为最大化支持，并且已知是 NP-hard。我们已经为它开发了几种有效的算法，这些算法提供了理论上的保证（常数因子的近似值），也有很好的扩展性，并在我们的测试网上成功测试。要了解更多信息，请参见我们关于 NPoS[12]中验证人选举的论文。

4.2 中继链状态机

从形式上看，Polkadot 是一个可复制的分片状态机，其中分片是平行链，而 Polkadot 中继链是协议的一部分，它确保所有平行链之间的全局共识。因此，Polkadot 中继链协议本身可以被视为一个可复制的状态机。在这个意义上，本节通过指定管理中继链的状态机来描述中继链协议。为此，我们说明了中继链状态，以及由中继链块中的交易分组所支配的状态转换的细节。

状态：状态是通过使用一个关联数组数据结构来表示的，该结构由（key，value）对的集合组成，每个键都是唯一的。除了 key 和 value 都必须是有限字节数之外，对 key 的格式和存储在 key 中的 value 没有任何前提限定。

构成中继链状态的(key, value)对排列在 Merkle radix-16 树中,这棵树的根可以有效识别中继链的当前状态。Merkle 树还提供了一种有效的方法来产生状态中单个配对的包含证明。

为了控制状态大小，中继链状态仅用于促进中继链操作，比如质押和识别验证人。Merkle Radix 树不会存储有关平行链内部操作的任何信息。

状态转换：像任何基于交易的转换系统，Polkadot 的状态变化是通过执行有序的指令集，即所谓的 extrinsics。这些 extrinsics 包括由公众提交的事务。它们涵盖了从机器状态的"外部"提供的任何数据，可以影响状态转换。Polkadot 中继链分为两个主要部分，即"Runtime"和"Runtime 执行环境"。状态转换功能的执行逻辑主要封装在"Runtime "中，而所有其他的通用操作，通常在现代基于区块链的复制状态机中共享，被嵌入到"Runtime 执行环境"中。特别是，后者负责网络通信、区块生产和共识引擎。

Runtime 函数被编译成一个 WebAssembly 模块，并作为状态的一部分存储。Runtime 执行环境将外部信息传递给 Runtime 并与其交互以执行状态转换。通过这种方式，状态转换逻辑自身就可以作为状态转换的一部分进行升级。

Extrinsics：Extrinsics 是提供给 Polkadot 中继链状态机以使其转换到新状态的输入数据。Extrinsics 需要被存储到中继链块中，以便在状态机之间实现共识。Extrinsics 分为两大类，即：交易和“inherents”，它们代表中继链块固有的数据。区块的时间戳 t 即为一个必须包含在每个 Polkadot 中继链块中的固有 extrinsics 示例。

交易被签名并在节点之间的网络上被广播。相比之下，inherents 不会被签名，也不会被单独广播，除非它们被包含在了一个区块中。如果绝大多数验证人都认

为这个 inherents 是有效的，那么一个区块中的 inherents 也会被假定为有效。中继链上的交易主要涉及中继链和 Polkadot 协议的整体操作，例如 set code, transfer, bond, validate, nominate,vote 等操作码。

中继区块生产者监听网络上所有交易消息,收到交易信息后，Runtime 验证其有效性，然后有效的交易会根据其优先级和从属性被排列在队列中，并相应地被考虑包含进未来的区块中。

中继链区块格式：一个典型的中继区块由 header 和 body 组成。body 由一系列 extrinsics 组成。

header 包含父块的哈希值、块号、状态树的根、Merkle 树的根（通过将 extrinsics 排列在 merkle 树中获得）以及摘要。摘要会存储来自共识引擎的辅助信息，这些信息将用来验证区块的有效性以及区块的来源，同时帮助轻客户端在无需访问状态存储的情况下验证区块。

构建中继链区块：在本节中，我们总结了由中继链验证人执行的中继链操作的各个步骤。通常，每个验证人都知道它什么时候应该生成一个块的（见 4.3.1）。

同时，从包括平行链区块链哈希、转换、质押、提名或因违反协议而受到惩罚的交易都会被提交给中继链验证人。验证人验证事务的有效性并将它们存储在事务交易池中。一旦预计验证人生成区块的时间段到达，验证人就会估算出最可能被终态协议确认状态的区块，并将其设置为中继链的当前状态。然后验证人从交易池中挑选有效交易进行执行，并相应地更新其状态。验证人在区块容量允许的范围内，执行并校对尽可能多的交易，并在执行所选交易之后，附加一个中继链最新阶段的加密摘要。最后，验证人签名并发布构建的中继链区块。

收到新区块后，其他验证人会检查区块生产者对协议的遵守情况以及所包含交易的有效性，并将该区块存储在区块树中，区块树中包含了所有可能成为中继链最终状态转换的候选人区块。

同时，验证人对区块树的各个分支（见 4.3.2 节）进行投票，并删除掉与绝大多数验证人认定的版本相冲突的分支。这样，他们最终就中继链的状态达成了一致。

4.3 共识

在本节中，我们将解释 Polkadot 的混合共识协议，该协议由 BABE 和 GRANDPA 组成，前者是中继链的区块生产机制，提供概率最终性，后者提供可证明的确定性最终性，独立于 BABE 工作。通俗的说，概率性终态即经过一定时间后，中继链中的一个区块将以非常高的概率（接近 1）得到终态确定，但可能后期不会被纳入获得大部分验证人一致认可的中继链分支；而确定性终态意味着一个被确定了的区块将永远保持确定性。此外，可证明的终态意味着我们可以向没有积极参与共识的各方证明一个区块是终态化的。

可证明的终态性将使转接桥更容易连接 Polkadot 之外的链。拥有和 Polkadot 不同共识的其它区块链，需要确信何时能安全地和中继链区块或平行链区块中的数据进行交互而没有任何被回滚的风险。保证这一点的最佳方法是，使处于中继链状态及平行链状态中的验证人都遵循拜占庭协议。另一方面，可用性和有效性机制 （第 4.4.2 节）可能要求我们能回滚区块，这意味着在每个区块上使用拜占庭协议（如 Tendermint[8]或 Algorand[23]）是不合适的，如果这样做，那么大量的

质押会被削减，所以这种情况应该尽可能少的发生。因此，我们想要一个能生成区块并乐观执行的机制，且一段时间后再确定它们的终态。因此，GRANDPA 选民在投票最终确定该区块终态之前，需要等待确保该区块的可用性和有效性被验证的证明。尽管我们敲定区块终态的速度可能有差异——如果我们没有收到区块无效和不可用的报告，那么我们可以快速确定它的终态，但如果我们收到区块无效和不可用的报告，那么在执行更多相关检查时，我们可能需要延迟终态确定。

由于 Polkadot 的跨链信息传递协议（XCMP 4.4.3）的工作方式，消息传递速度受出块时间的限制，但不受终态时间的限制。因此，如果我们延迟终态确定并最终不回滚，那么消息传递仍然会很快。

基于这些要求，我们选择尽可能地将区块生成机制和确定区块终态的机制分开。在接下来的两节中，我们将分别描述执行这些操作的 BABE 和 GRANDP 协议。

4.3.1 BABE

在 Polkadot 中，我们使用 BABE (Blind Assignment for Blockchain Extension) 协议来生成中继链区块。BABE 利用父区块产生时生成的随机数，将验证人随机分配到下一个 slot（一个区块生成的时间，6 秒）中。这些分配是完全不公开的，直到分配的验证人生成他们的区块。因此，我们在协议名称中使用“Blind Assignment”。BABE 类似于 Ouroboros Praos [15]，但它们在链选择规则和时间假设方面存在一些显著差异。

在 BABE 中，可能存在没有分配到任何验证人的 slot，我们称之为空 slot。为了填补空 slot，我们有一个基于 Aura [31]的辅助区块生产机制，该机制会公开的将验证人分配给空 slot。然而，由于最佳链选择规则和随机数生成算法，会默认 Aura 机制生成的块不存在，所以基于 Aura 生成的区块对 BABE 的安全性没有贡献。接下来我们仅将 BABE 与其安全性一起描述。

BABE[2]由一个个称为 epochs($e_1, e_2,...$)的时间分隔组成，其中每个 epoch 由多个连续的 slot 组成 ($e_i = \{sl_1^i, sl_2^i, ..., sl_t^i\}$)，网络可以设定的每一个 epoch 的 slot 数目上限 R。在每个 epoch 开始时，每个验证人都知道它应该在其中哪个 slot 中生成一个区块。当属于它的 slot 到来时，验证人通过证明它已分配给该 slot 来生成区块。

盲分配基于可验证随机函数 (VRF) [24]的加密原语（参见第 4.7.2 节）。一个 epoch e_m 中的某个验证人可执行如下操作，以了解它是否有资格在 slot 的sl_i^m中生成区块：

1. 如果m=1 或m=2，则 BABE 获取创世区块中的随机性；否则，需要获取 (e_{m-2}) 之前的两个 epoch 生成的随机性；

2. 它使用秘密私钥和输入运行 VRF：随机性和 slot 号sl_i^m。

如果 VRF 的输出小于阈值T，则这个验证人领导该 slot，意味着它有资格为该 slot 生成区块。我们根据 BABE[2]的安全性要求选择 T，例如，与较小的 T 相比，较大的 T 使降低了在一个 slot 中被选择的验证人全是诚实的可能性。当验证人产生一个区块时，它会将 VRF 的输出和证明添加到该区块中，表明其 VRF 输出小于 T，以向其它验证人说明它有权在相应的 slot 中产生一个区块。验证人总是在最佳链上生成它们的区块。BABE 中的最佳链选择规则是忽略 Aura 区块的

同时选择一条最长链，该最长链包含了最后一个由 GRANDPA 确定了终态的区块。见章节 4.3.2 关于如何在 GRANDPA 中确定区块终态的细节。

一个 m > 2 的 epoch e_m 的随机性是通过使用属于该 epoch 的最佳链的 BABE 块生成的：令 p 是属于 e_m 的 BABE 块中所有 VRF 值的参数；然后，以 $r_m = H(\mathrm{m}\|\mathrm{p})$ 来计算 epoch e_m 的随机性，其中 H 是一个哈希/散列函数。验证人定期运行下面描述的相对时间算法，以根据验证人的本地时钟了解 slot 在什么时间开始。

相对时间协议：分配给一个 slot 的验证人需要知道什么时候是正确的区块产生时间，以保护 BABE 的一致性和安全性。为此，验证人使用他们的本地计算机时钟，该时钟不受任何中心化时钟调整协议（例如网络间协议[25]）的校正。相反，他们使用相对时间协议使时钟与其他验证人同步。关于在没有 NTP 的区块链场景中本地时钟同步的正式安全模型，以及找到有关相对时间协议的更多详细信息，可以在文献[5]中找到说明。

在 BABE 中，我们假设在释放创世区块之后，在第一个 epoch 被选中的验证人负责存储创世区块的到达时间以及其相应的本地时钟。然后，他们标记第一个 slot 的开始时间，并每 T 秒增加一次 slot 数。此后，他们会定期运行相关算法，以免因为本地时钟推移而失去与其它验证人的同步。除此之外，在创世区块之后运行相对时间算法的验证人需要与其他验证人同步。

在每个 sync-epochs（不同于 BABE 中的 epoch），验证器根据相对时间协议的结果更新他们的时钟，并使用新时钟直到下一个 sync-epochs。第一个 sync-epochs 在创世区块被释放后立即开始。当最后一个（概率上）被确定了终态的区块的 slot 号是$\overline{sl}_{\varepsilon}$时（$\overline{sl}_{\varepsilon}$是最小的 slot 号），另一个 sync-epochs 开始。这使得 $\overline{sl}_{\varepsilon}$-$\overline{sl}_{\varepsilon-1} \geq s_{cd}$，其中$\overline{sl}_{\varepsilon-1}$是$\mathrm{sync-epoch}_{\varepsilon-1}$中最后一个（概率上）确定了终态区块的 slot 号。这里，s_{cd}是链密度（CD）性能参数，它将根据链的增长来定义。更详细地，每个验证人在 sync-epoch 期间将块的到达时间t_j与块中的 slot 号sl'_j一起存储。在 sync-epoch 结束时，验证人检索 sync-epoch 期间生成的概率性终态区块的到达时间，并计算下一个 sync-epoch 的第一个 slot sl 中一些候选人的开始时间，即$a_j = T(sl - sl'_j), C_T = \{t_j + a_j\}$。$C_T$中的时间被视为候选人时间集合。为了选择一个候选人，验证人对候选列表C_T进行排序，并输出排序列表的中值作为 sl 的开始时间。图 3 是在第一个 sync-epoch 中执行相对时间协议的示例。

BABE 的安全性概述：为了获得安全的区块链协议，Garay et al. [17]定义了下面的属性。非正式地，我们可以通过以下属性对其进行总括描述这些属性:

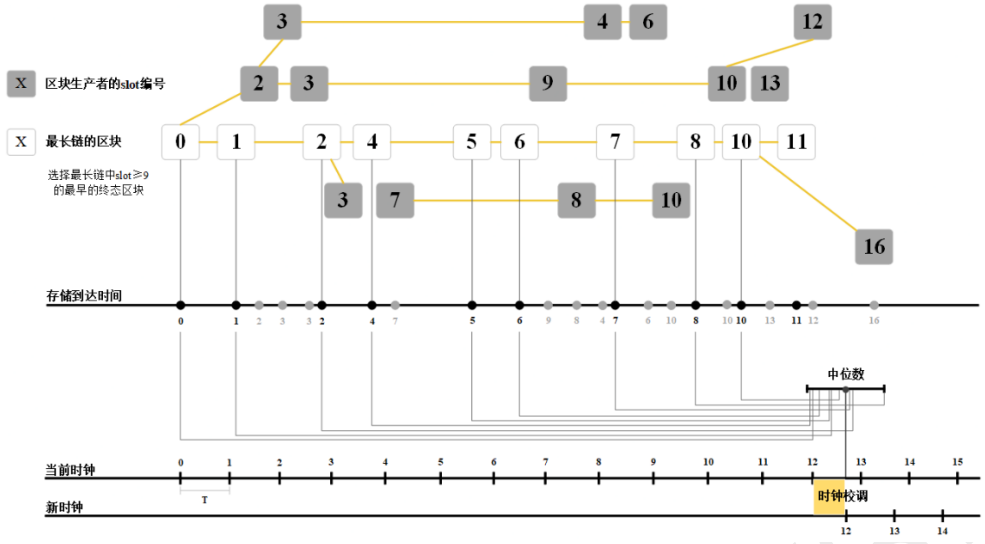

图 3：当 s_{cd}=9 时，第一个 epoch 内相对时间协议执行流程示意

- 通用前缀(CP, Common Prefix)：它确保在一个拥有诚实验证人的区块链的最后一个区块之前的 k 个区块不能被更改。我们称所有不可更改的块为最终块。由于一个 slot 中分配到恶意验证人的概率远低于分配到诚实验证人的概率，所以在一个 slot 中绝大多数验证人是诚实的，因此 BABE 满足 CP 属性，这意味着，恶意验证人没有足够的能力去构建一条不包含任何一个最终区块的链。
- 链质量(CQ, Chain Quality)：它确保在每一个确定的 slot 中，一个最小诚实区块可为一个诚实方拥有的最佳链做出贡献。即使在最坏的情况下，即网络延迟最大的情况下，我们也可保证在一个 epoch 内最佳链中至少会有一个诚实的区块，这样随机性不会有偏差。
- 链增长(CG, Chain Growth)：它保证 slot 之间的最小增长。由于绝大多数诚实验证人的存在，使得恶意验证人无法阻止最佳链的增长。
- 链密度(CD, Chain Density)：它确保在最佳链的足够长的部分中，有一半以上的区块由诚实的验证人产生。CQ 和 CG 影响这个属性[15]。

有关 BABE 及其安全性分析的更多详细信息，请参见[2]。

4.3.2 GRANDPA

如上所述，我们希望有一个灵活的、与区块生产分离的最终确定机制，这一点由 GRANDPA 实现。为了与 GRANDPA 一起工作，对 BABE 的唯一修改是改变分叉选择规则：验证人生成的区块不是建立在最长的链上，而是建立在被敲定的最终完成的最长的链上。GRANDPA 可以与许多不同的区块生成机制一起工作，并且有可能用另一种机制来取代 BABE。

直观地说，GRANDPA 是一个拜占庭协议，其功能是从许多可能的分叉中就一条链达成协议。该功能主要通过以下两方面实现：一，遵循部分更为简单的分叉选择规则；二，即便 GRANDPA 本身停止最终区块敲定区块，其区块生成机制也能够根据概率决定最终区块。我们希望能够同时就许多新生成的区块达成共识，这与单区块拜占庭协议有所不同。

我们假设，我们可以向分叉选择规则询问给定的最佳区块。基本的想法是，我们想在大家都同意的链的前缀上达成拜占庭共识。为了使其更加具备健壮性，我们试图就 2/3 的验证人同意的链的前缀达成一致。

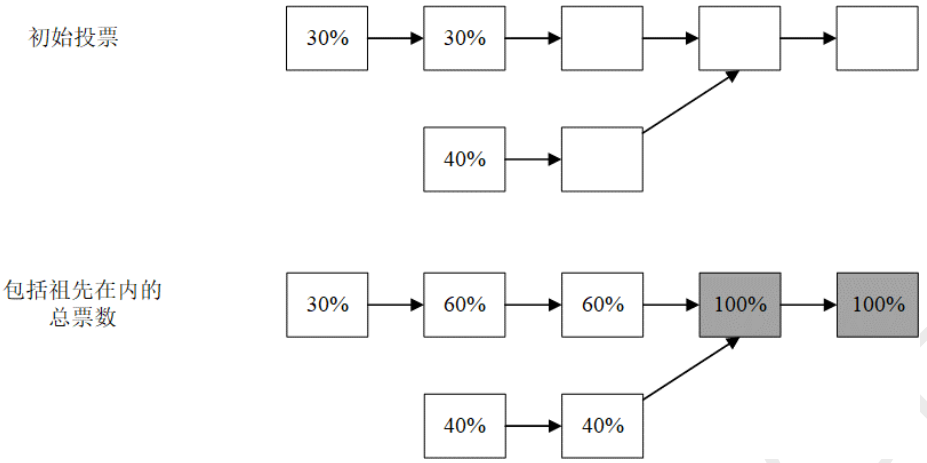

图 4：GRANDPA 票数及其汇总方式

我们在投票规则上使用了贪婪最重可观测子树（GHOST）算法，很像 Casper TFG 或一些建议用于 Casper FFG 的分叉选择规则。我们的内部结构上像一个更传统的拜占庭协议形式来使用这个规则处理投票。如图 4 所示，GHOST 规则工作原理如下：我们设置一组由区块哈希给出的投票，诚实的验证人有且只有一票，并去由以下方式归纳形成的链头。我们从创世区块开始，然后包括该区块中 2/3 的投票者投票给其后生成的区块，只要正好有一个这样的即可。这个区块头包含 g($\mathcal{V}$)，其中$\mathcal{V}$是投票的集合。如图 4 所示，左边是单个区块的票数，右边是每个区块及其所有后代的总票数。创世区块位于顶部，我们从它的子块中选出得票率为 100%（>2/3）的票数。该区块的子代分别拥有 60%和 40%的票数。因为这些票数低于 2/3，所以我们停止并返回第二个区块。

一轮 GRANDPA 共识有两个投票阶段：预投票和预提交。首先，验证人对最佳链进行预投票；然后，他们根据 2/3-GHOST 规则，g，应用于他们所看到的预选票集$\mathcal{V}$并对 g($\mathcal{V}$)进行预提交；最后，他们对看到的预提交集合$\mathcal{C}$和最终确定 g($\mathcal{C}$)。

为了确保安全，我们确保所有的投票都是有可能在上一轮被敲定的任何区块的后代。节点保持着对上一轮可能已经完成的区块的估计值，该估计值是根据预投票和预提交中计算出来的。在开始新的一轮之前，一个节点会等待，直到它看到足够多的预提交，以确保在不同的链上或后来的同一链上没有区块可以最终完成本轮的区块敲定。然后，它确保在下一轮只对属于上一轮估计的后代区块进行预投票和预提交，它通过监听上一轮的预承诺来持续更新。这就保证了安全性。

为了确保有效性，我们轮流选择一个验证人作为主验证人。他们在这一轮开始时广播他们对上一轮的估计值。然后，当验证人进行预投票时，如果主块通过了两项检查，即它至少是验证人的评估结果，并且在上一轮中它和它的后代得到超过了 2/3 的预投票，那么它就会对包括主块在内的最佳链进行预投票。这里的想法是，如果主块没有被最终敲定，那么通过最终敲定区块来取得进展。如果主块没有被最终确定，并且所有验证人都同意包括最后一个最终确定的区块在内的最佳链，那么我们最终应该这样做。因为 BABE 给出了概率上的最终确定，那么我们现在通过最终确定该链取得进展。

4.4 平行链

在这一节中，我们将了解平行链区块的生成，它们的可用性和有效性方案，以及他们的信息传递方案。

4.4.1 区块生成

我们将讨论通用平行链的区块生成过程。在本节的最后，我们将讨论其备选方案。

概括地说，收集人产生一个平行链区块，将其发送给平行链验证人，验证人对区块头进行有效签名，有足够签名的区块头将被放在中继链上。在这一点上，平行链区块和它的头出现在中继链上的块是一样的。根据 BABE（见第 4.3.1 节），这个中继链区块在最佳链中，那么平行链区块也在最佳链中，当这个中继链区块被最终确定时，平行链区块也在最佳链中。

因为平行链验证人经常切换到不同的平行链，它们是平行链的无状态客户端。因此，我们区分了平行链区块 B 和有效性证明（PoV）区块 B_{PoV}，前者通常能够让平行链完整节点如收集人更新平行链状态，后者则使不具备平行链状态的验证人能够验证。

任何验证人都应该能够在给定中继链状态的情况下，使用平行链的状态转换验证功能（STVF）来验证 B_{PoV}，其 WASM 代码以类似中继链 runtime 的方式存储于中继链上。STVF 将 PoV 区块作为输入，包含从该平行链最新的区块头和中继链状态中的少量数据。

STVF 输出该区块的有效性，该区块头和它所发出的信息。PoV 区块包含任何出站信息和平行链区块 B。平行链验证人应该将该平行链区块向平行链网络进行广播，正如收集人自己所做的工作。

PoV 区块会以平行链区块形式存在，它的出站信息、区块头和轻节点客户端见证人。这些见证人节点能够运行默克尔根证明，即给出了输入和输出状态的所有元素，这些元素能被输入和输出状态根的状态转换函数所使用或修改。

为建立无审查网络，一条平行链可能采用 PoW 或 PoS 机制来选择收集人，而如何选择收集人机制则有平行链自行决定。这一点可以在 STVF 中自行实现，而无需成为 Polkadot 协议的一部分。例如对 PoW，STVF 将检查区块哈希值是否足够小。然而，为了提高 TPS，确保大多数中继区块中可以包含一个平行链区块是很有用的。对于 PoW 来说，可能需要允许多个收集人产生一个区块。因此，我们需要一个有效解决的方法，让平行链验证人在验证同一个平行链区块之前先进行协调。这可能是异构多链框架愿景中的金票计划。当然，对于 PoS 来说，这可能没有必要。

另外，对于某些平行链来说，平行链区块 B 可能不足以让收集人更新其状态。这可能发生在使用简洁的零知识证明来更新其链上状态，或者甚至发生在只提供权威机构签名的许可链上的有效性。这样的链可能有其他概念，即实际需要更新其状态的平行链区块，必须有自己的方案来保证这些数据的可用性。

4.4.2 可用性和有效性

一旦创建了平行链区块，重要的是由 PoV 块和平行链的出站消息集组成的平行链 blob 在一段时间内是可用的。乐观的解决方案是向所有中继链节点广播平行链 blobs，这不是一个可行的方案，因为有很多平行链和 PoV 块可能很大。我们希望找到一个有效的解决方案，以确保任何最近创建的平行链区块中 PoV 块是可用的。

对于独立的链，如比特币，只要 51%的哈希值是诚实的，不提供区块数据就能确保没有诚实的矿工在其上生产区块，所以它将不会出现在最终的链上。然而，Polkadot 中的平行链间共识是由中继链共识决定的。当一个平行链区块头在中继链上时，这是典型的形式。我们不能保证除收集人和平行链验证人之外的任何其他人都看到 PoV 块。如果这些人串通好了，那么其他的平行链网络就不需要有平行链区块，那么大多数收集人无法构建新区块，这个区块的无效性可能不会被发现。我们希望共识的参与者，也就是验证人，能够共同确保可用性，而不是依靠少数节点。

为此，我们设计了一个可用性方案，使用擦除编码（见附录 4），将 PoV 块分配给所有验证人。当检测到任何不当行为，特别是与无效性有关的行为时，可以从分布式的擦除编码的块中重建该 blob。

如果一个区块是可用的，那么平行链的所有节点，以及任何拥有 PoV 的轻客户端，都可以检查其有效性。我们在 Polkadot 中有三个级别的有效性检查。PoV 块的第一个有效性检查是由相应的平行链验证人执行的。如果他们验证了 PoV 块，那么他们会签署并分发 blob 的擦除代码，包括 PoV 块，给每个验证人。我们依靠充当钓鱼节点来报告一个 blob 的无效性，作为第二层的有效性检查。他们需要质押自己的 DOT 来支持任何主张。我们假设大多数收集人都是钓鱼人，因为他们对链持续有效性有兴趣，并且已经在运行完整的节点，所以他们所需要是 DOT 质押收益。第三层的有效性检查是由一些随机和私人分配的验证人执行的。考虑到钓鱼人提供的无效报告和收集人提供的不可用报告的数量，我们确定第三级有效性检查中的验证人数量。如果检测到一个无效的平行链区块，为其有效性签名的验证人就会被砍掉。我们等待足够多的这些随机分配的检查者检查该区块，然后在 GRANDPA 中对其投票。我们还希望在选择随机分配的验证人之前，确保该区块是可用的。这意味着平行链验证人必须承诺因为得到一个无效区块的小概率而承担被惩罚的高风险。这意味着让一个无效区块进入 Polkadot 的预期成本高于质押单个平行链的收益。

我们的可用性和有效性方案的安全性取决于 GRANDPA 最终性确定装置的安全性（见第 4.3.2 节）和每个 BABE epoch 中产生的随机性的质量（见第 4.3.1 节）。关于可用性和有效性方案的更多细节，请参见附录 A。

4.4.3 跨链消息传递（XCMP）

XCMP 是平行链用来相互发送消息的协议。它旨在保证四件事：第一，信息快速到达；第二，从一个平行链的信息按顺序到达另一个平行链；第三，在发送信息的平行链历史数据中，到达的信息确实已被发送；第四，接收者将公平地收到不同发送者的信息，帮助保证发送者不会无限期地等待其信息被看到。

XCMP 有两个部分。(1) 一个平行链区块的发送信息的元数据被包含在中继链上，随后这些元数据被接收平行链用来验证信息。(2) 与该元数据相对应的信息体需要从发送者实际分发到接收者，同时还要证明该信息体确实与相关元数据相关。分发的细节作为网络协议在跨链消息中有所涉及；其余部分在下文中介绍。

中继链块包括平行链区块头的方式给了平行链区块一个同步的时间概念，通过中继链块的编号。此外，它允许我们在中继链给出的历史中验证消息的发送，也就是说，不可能出现一个平行链发送了一个消息，然后重新排序，使该消息没有被发送，而是被收到。即使系统可能没有对信息是否被发送达成最终结论，这也是成立的，因为任何中继链都提供了一个一致的历史。

因为我们要求平行链最终对每条消息采取行动，不交付一条消息就有可能使平行链无法建立区块。因此，我们的消息传递系统需要足够的冗余度。任何验证 PoV 区块的验证人都应该将该区块的任何发送消息保留一天左右，并且发送区块的所有全节点也会储存外发消息，直到他们知道它们已经被采取行动。

为了实现一致性，当作为发送源的平行链 S 向接收方平行链 D 发送平行链 B 中的消息时，那么我们就需要使用中继链状态来验证这些消息，而中继链状态是根据中继链中包含的平行链 B 对应的平行链区块头 PH 来更新的。我们需要在中继链状态中限制像 PH 这样的头文件的数据量，同时也要限制中继链在处理这种平行链头文件时需要做的认证工作。

为此，平行链头 PH 包含一个发送消息的消息根 M，以及一个表明该区块中哪些其他平行链被发送消息的位域。消息根 M 是该区块发送消息的每个副链 p 的头哈希 H_p 的 Merkle 树的根。头 H_D 的哈希链有所有从 D 发送至 S 的信息的哈希值，不仅仅是在块 B 中，而是在块 B 之前的任何块中从 S 发送至 D 的信息。这允许从 S 到 D 的许多消息依次从 M 处被验证。无论消息本身如何被传递，它们也应该与 Merkle 证明一起被发送，该证明允许接收平行链的节点认证它们是由头 P_H 在特定中继链块中的 B 发送的。

平行链 s 按顺序接收传入的消息。在内部，平行链 s 可以根据他们自己的逻辑（可能受到 SPREE 的约束，见 A.1）推迟或重新安排对消息的作用。然而，他们必须按照中继链给出的一致历史所决定的顺序来接收消息。一个平行链 D 总是先接收由其头在早期中继链块中的平行链块发送的消息。当几个这样的源平行链在中继链块中有一个区块头时，来自这些平行链的消息会以某种预定的平行链顺序被接收，可以按平行链 id 增加的顺序，也可以是某种洗牌的版本。

一个平行链 D 接收一个平行链 S 在一个平行链块中发送的所有信息，或者不接收任何信息。D 的一个平行链头 PH’包含一个水印。这个水印由一个中继链块 R 的块号和一个源平行链 S 的平行链 id 组成。这表明 D 已经收到了中继链块 R 之前的所有链所发送的所有消息，并且在排序中对 R 块中由平行链（包括 S）发送的消息进行了操作。

水印必须在 D_s 的每个平行链块中至少领先一个发送平行链，这意味着水印的中继链块数超前或保持不变，我们只超前平行链。如果要在建立在某一特定中继链块 R 上的平行链 D 上产生一个平行链区块，收集人需要查看该链的最后一个平行链块所建立的中继链块之间有哪些准链头。此外，它还需要每一个表明他们向 D 发送消息的人的相应信息数据。因此，它可以构建一个 PoV 块，以便 STVF 可以验证所有这些消息被采取行动。由于一个平行链必须接受所有发送给它的消息，我们为平行链实现了一种方法，使另一个平行链向它发送的任

何消息都是非法的，这些消息可以在发生垃圾邮件的情况下使用。当发送消息的平行链块的头被包含在中继链块中时，那么任何连接到源和目的平行链网络的节点都应该将消息连同其证明从发送者转发给接收者。中继链至少应该起到备份作用：D 的接收方平行链验证人与 D 的平行链网络相连，如果他们没有在该网络上收到消息，那么他们可以在消息发送时向发送链 S 的平行链验证人索要消息。

4.5 经济激励层

Polkadot 有一个为 DOT 的原生 Token。它的各种功能将在本节中描述。

4.5.1 质押奖励和通胀设计

我们首先描述一下质押奖励，即对质押人的收益支付--验证人和提名人来自新铸造的 DOT。与其他一些区块链协议不同，Polkadot 中的 Token 数量不会被一个绝对的常数所限制，而是会有一个可控的年度通货膨胀率。事实上，最近的研究表明，在一个基于 PoS 的协议中，质押奖励必须保持竞争力，以维持高质押率和高安全水平，所以建议不要采取通货紧缩政策。

在我们的设计中，质押奖励是开采 DOT 的唯一机制。因此，在这一节中介绍我们的通货膨胀模型也很方便。

从 NPoS 协议的描述（第 4.1 节）中可以看出，验证人和提名人都持有 DOT。他们得到的报酬与他们的股份大致成正比，但在行为不端的情况下，可以削减到 100%。尽管他们每次只积极从事一个 era 周期工作，但他们可以继续从事无限数量的 era。在此期间，他们的股权被锁定，这意味着他们不能花钱，而且在他们最后一个活跃的 era 之后的几个星期内，他们的股权仍然被锁定，以便即使犯罪行为被发现的时间较晚，也要对作恶者进行处罚。

抵押率，利率，通货膨胀率：让质押率成为当前由验证人和提名人押注的 DOT 总量，除以当前的 DOT 供应总量。质押人的平均利率将是质押率的一个函数：如果质押率低于治理部门选定的某个目标值，平均利率就会提高，从而激励更多的人参与 NPoS，反之亦然。例如，可以选择 50%的目标押注率作为安全和流动性的中间地带。如果在这个水平上，质押人的平均年利率被设定为 20%，我们可以预期通货膨胀率将在 50%×20%=10%左右紧密波动。因此，通过设定质押率和质押利率的目标，我们也控制了通货膨胀率。遵循这一原则，每个 era 我们都会调整我们对押注率的估计，并利用它来计算该 era 周期要支付给质押人的 DOT 总额。

跨验证器支持的奖励：一旦计算出当前 era 的总报酬，我们就需要确定它是如何分配的。回顾一下，验证人选举协议（第 4.1 节）将有效股权划分为验证人支持，其中每个验证人支持由一个验证人的全部股权加上其支持的提名人的部分股权组成，这种划分是为了使验证人支持尽可能高且均匀分布，从而确保安全和分散化。为确保权力下放而建立另一个激励机制，是为同等的工作支付验证人的支持，无论他们的质押数量多少。因此，如果一个受欢迎的验证人有很高的支持率，它的提名人可能会比支持不那么受欢迎的验证人的提名人在每 era 周期 DOT 上得到更少的报酬。因此，提名人将被激励改变他们的偏好，以

支持不太受欢迎的验证人（虽然有良好的声誉），帮助系统收敛到所有验证人支持有同等股权的理想情况。

特别是，我们设计了一个积分系统，其中验证人每完成一个收费的动作就会积累积分，在每个 era 周期结束时，验证人的槽位按其积分比例得到奖励。这确保了验证人总是被激励去保持高性能和高响应性。Polkadot 的可支付行动包括：a）验证一个平行链区块，b）在 BABE 中产生一个中继区块，c）在 BABE 区块中加入对一个先前未引用的叔块的引用，和 d）产生一个叔块。

验证人卡槽内的奖励：由于提名人的质押通常被分成几个验证人的支持，他们在一个 era 内的收益相当于他们每个支持的收益之和。在一个验证人支持中，支付情况如下：首先，验证人被支付一笔佣金，这是一个旨在支付其运营成本的金额；然后，余下的部分由所有质押人--包括验证人和提名人--按其股权比例共享。因此，验证人收到两个独立的奖励：运行节点的费用和质押的收益。我们注意到，佣金费用是由每个验证人自己设定的，而且必须事先公开宣布。较高的费用意味着验证人的总报酬较高，而对其提名人的报酬较低，因此提名人一般会倾向于支持费用较低的验证人，而市场在这方面会自我调节。然而，那些在可靠性和业绩方面建立了良好声誉的验证人将能够收取更高的佣金费用，这也是公平的。

在本节的最后，我们对我们的支付方案预计会对投票者产生的激励进行了一些观察总结。首先，由于验证人报酬丰厚且数量有限，他们有动机确保来自提名人的高额支持以确保当选，因此他们会重视自己的声誉。随着时间的推移，我们预计选举将是高度竞争的，当选的验证人将有强大的业绩和可靠性记录，以及大量的股权支持；其次，即使不同验证人支持的报酬与他们的质押无关，但在一个验证人支持中，每个行为人的报酬与他们的质押成正比，所以总会有个人动机来增加自己的质押；最后，如果一个验证人获得了特别高的支持，它可以通过增加其佣金费用来从中获利，这样做的效果是在失去一些提名的风险下提高自己的回报，或者推出一个新的节点作为验证人候选人，并在其所有节点之间分割其支持。在这最后一点上，我们欢迎拥有多个验证人节点的运营商，甚至旨在使他们的物流更简单。

4.5.2 中继区块限制和交易费设计

对资源使用的限制：我们对一个中继区块所能处理的交易量进行了限制，目的是：a）确保每个区块即使在性能较差的节点上也能有效处理，避免区块生产的延迟；b）即使在网络流量很大的情况下，也能保证一定数量的高优先级、业务交易（如不当行为报告）的可用性。特别是，我们对以下资源设置了区块约束：链上字节长度，以及处理交易所需的时间和内存。

我们根据事务的优先级和资源消耗情况，将其分为几种类型。对于这些类型中的每一种，我们都根据最坏情况下的状态以及不同的输入参数进行了测试。从这些测试中，我们建立了每个事务的资源使用量的保守估计，我们使用这些估计来确保所有的资源使用限制得到遵守。

我们还增加了一个额外的资源约束：用来区分普通交易和高优先级交易，只让普通交易占到每个区块资源限制的 75%。这是为了确保每个区块的高优先级交易至少有 25%的资源保证空间。

交易费用：我们使用上述模型，根据三个参数设定交易费用水平：交易类型、链上长度以及预期资源使用量。这种费用的区分是用来反映一个交易在不同网络和地区环境中产生的不同成本，并鼓励处理某些类型的交易而不是其他。每笔交易费用的一部分被支付给区块生产者，而另一部分则被用于资助国库（第 4.6.4 节）。我们强调，对于区块生产者来说，来自交易费的奖励可能只占其总体收入的一小部分，只足以激励其加入区块。

我们还运行一个自适应的交易费用计划以应对不同流量状况，并确保日常区块避免满载的情况，因此活动的高峰期可以得到有效处理，并最小化尖峰时刻的出现概率。特别是，每笔交易的费用都乘以一个参数，该参数根据当前的网络流量随时间变化而变化。

我们使交易费的发展足够缓慢，所以任何交易的费用都可以在一个小时的框架内准确预测。特别是，我们并不打算让交易费成为验证人的主要收入来源。

4.6 治理

Polkadot 使用复杂的治理机制，使其能够随着时间的推移在其集合的利益相关者的最终要求下优雅地发展。一个关键和不变的规则是，对协议的所有修改必须由利益相关者加权公投同意——确保多数股权总能控制网络发挥作用。

为了对网络进行任何改变，我们的想法是将 DOT 持有人聚集在一起，在理事会的帮助下管理一个网络升级的决定（见第 4.6.2 节）。无论提案是由 DOT 持有者还是理事会提交的，最终都必须通过全民投票，让所有 DOT 持有者按利益加权，做出决定。

波卡的每个 DOT 持有者都有权：a）提交提案；b）认可一项公共提案，以便在公投时间表中优先考虑；c）对所有正在进行的公投进行投票；d）成为理事会席位的候选人；以及 e）对理事会候选人进行投票。此外，任何 DOT 持有人都可以成为参与 NPoS 的提名人或验证人候选人（见第 4.1 节）。

4.6.1 提案和公投

Polkadot 逻辑的核心是在链上存储在一个无定形的状态转换函数中，并以一种平台中立的语言定义：WebAssembly。每个提案都以 runtime 的特权函数调用的形式出现，能够修改 runtime 的代码本身，实现无缝升级，避免出现"硬分叉"的情况。然后，一个提案被提交，并通过公投进行表决。

提案可以通过以下几种方式之一启动：

- 公共提案：任何 DOT 持有人均可提交；
- 理事会提案：由理事会成员提交；
- 自动提交的提案：由前轮公投的部分内容，作为提案自动递交；
- 紧急提案：由技术委员会提交（第 4.6.2 节）。

每项经公投批准的提案都有一个相关的颁布延迟，即公投结束和修改颁布之间的时间间隔。对于前两种类型的提案，这是一个固定的时间间隔，暂定为 28 天；对于第三种类型，它可以根据需要设置；紧急提案处理的是网络中的重大问题，需要快速处理，因此会有一个较短的颁布延迟。颁布延迟确保了一定

程度的稳定性，因为它给了所有各方足够的通知来适应新的变化。在这段时间之后，会自动调用相关的特权函数。

任何利益相关者都可以通过存入固定的最低数量的 DOTs 来提交公开提案，并在一定时期内保持锁定。如果有人同意该提案，他们可以存入相同数量的 Token 来支持它。公共提案被储存在一个优先级队列中，每隔一段时间，得到最多赞同的提案会被提交给公投。一旦提案被提交，锁定的 Token 将被释放。

理事会的提案由理事会提交，并存储在一个单独的优先级队列中，优先级由理事会自行确定。

公投是一种简单的、包容的、加权的投票方案。它有一个固定的投票期，投票结束后进行统计。公投总是二元的：投票选项是 "赞成"、"反对 "或完全弃权。

时间表：每隔 30 天，就会有一个新的提案被提出来，并进行公投。要提交的提案是公众提案队列或理事会提案队列中最重要的提案，如果两个队列都不空，则在这两个队列中交替进行。如果两个队列都是空的，则在公投时间表中跳过该时段。多个公投不能同时进行，但紧急公投除外，因为紧急公投的时间表是平行的。

计票：公投的投票对所有 DOT 持有者开放，投票权与他们的股份成正比，最多有一个可能的投票倍数，根据一些组织对系统的承诺程度，给与他们投票权。一般来说，一个组织必须锁定他们用于投票的 Token，至少要到公投结束后的颁布延迟期。这是为了确保需要对结果有一些最低限度的经济买入，并劝阻卖票。完全不锁定投票是可能的，但在这种情况下，投票权只是给定股权的正常投票的一小部分。相反，Polkadot 将提供自愿延长锁定，允许任何一方通过延长他们愿意锁定其 Token 的时间来增加他们的投票权。这确保了长期致力于该系统的选民，愿意增加他们对公投决定的接触，在这个问题上有更大的发言权。

投票率的偏差：以公共提案形式来要求所有利益相关者参与解决某些细小问题上似乎是不必要的。例如，要求出块时间降低 5%这类问题。然而，如果没有这个规则，网络很可能是不稳定的，因为把它的控制权放在利益相关者的手中之外，会产生一种错位，可能会导致不作为或更糟。然而，通过利用投票率很少是 100%这一事实，我们可以根据情况产生不同的结果，在主动和被动的利益相关者之间建立起权力平衡。例如，简单的投票系统通常会引入一个法定人数的概念，即必须达到最低的投票人数才能通过一项变革。

对于公共提案，我们将这一概念概括为"积极的投票率偏差"，即假设赞成与反对的比例相同，额外的投票率总是使变革更有可能。更具体地说，在投票率低的情况下，我们通过要求超级多数的批准来支持反对方或现状，而当投票率接近 100%时，要求就会降低到多数通过。这基于两个原则：首先，现状往往比任何变化都更安全，因此应该对现状有一些偏差；其次，像所有的经验测量手段一样，不可避免地会有一定程度的不准确性和波动性，特别是在投票率较低的情况下--结果可能在一个月内是 51%-49%，然后变为 49%-51%，考虑到颁布提案变化所涉及的成本，确保结果不可能在颁布后不久翻转是有利的。

另一方面，对于理事会提交的提案，全民投票没有投票率的偏差，并遵守多数通过的原则。这里的理由是：理事会预先批准的提案被认为更安全，更不

可能被推翻，所以之前提到的问题得到了缓解，我们可以让 DOT 持有人自由决定此事。

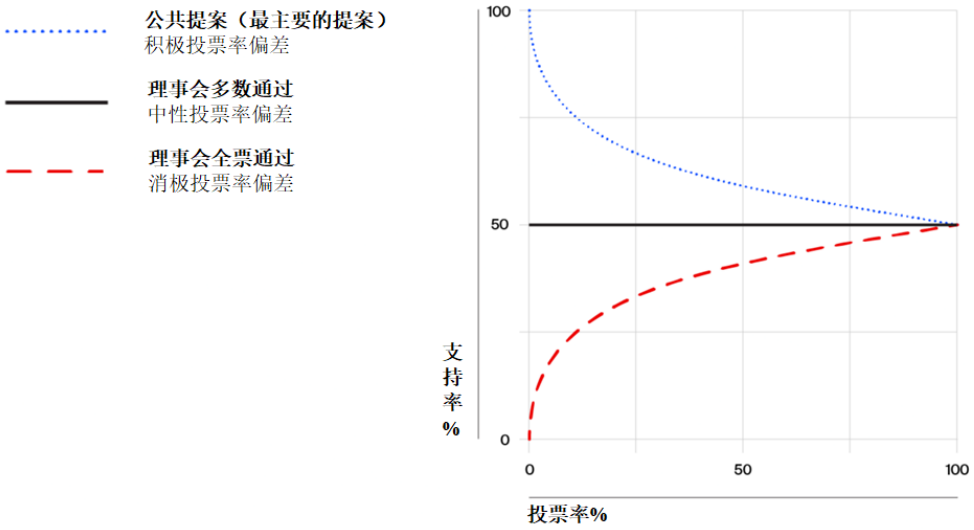

图 5：自适应投票率偏差

最后，在理事会提案得到所有理事会成员一致支持的特殊情况下，将观察到 "负投票率偏差"。这是第一种情况的对称反面，即额外的投票率总是使变化的可能性降低，我们在投票率低的情况下有利于赞成方，要求反对者的超级多数来拒绝提案，当投票率接近 100%时，要求调低为多数通过。见图 5。

4.6.2 理事会和技术委员会

理事会是一个由若干行为者组成的实体，每个行为者由一个链上账户代表。它的目标是代表被动的利益相关者，提交明智和重要的建议，并取消无争议的危险或恶意的建议。

理事会将不断审查候选提案，以处理系统中出现的问题。一个候选提案只有在得到绝大多数理事会成员的批准，并且没有成员行使否决权的情况下，才会得到理事会的正式支持，并进入理事会提案的队列。一个候选提案只能被否决一次；如果在冷静期过后，该提案再次被理事会的多数成员批准，则不能被第二次否决。

如前所述，得到全员支持票的提案，被认为是无可争议的提案，并且享有特殊的投票结果计算偏差数值，使其更有可能被批准。

最后，无论是提案是有谁提交的，理事会成员都可以在全体一致性投票规则下，行使取消任何提案的权利。由于要达到全体一致性的要求非常高，预计该措施只有在完全没有争议的情况下才会被使用。当天临近执行的提案出现运行代码错误或网络安全问题情况时，此类措施可作为最终备选措施来进行否决投票。如果否决决策存在一个及以上反对者，则说明存在争议，那么提案的决否权将留给所有 DOT 持有者决定提案的命运，此外，为引起投票者们的足够关注，同时会附上相关理事会投票否决取消提案的信息，以作为风险告知。

理事会成员选举：波卡理事会将有 23 个席位。所有席位的选举将每月进行一次。所有 DOT 持有者都可以递交候选人申请，并可以自由投票给其他候选人，其投票权重与其 Token 质押正相关。类似 NPoS 中的验证人选举，这是一

个基于投票批准的质押加权、多赢家的选举议题。因此，我们可以使用用于NPoS 的相同算法来解决这个议题，NPoS 提供了比例调整代表的特性。详细内容请参阅第 4.1 节。这个特性保证了被选举的理事将代表尽可能多的少数群体，从而确保治理的去中心化和避免被操纵。理事会成员可以无限期连选连任，前提是他们的支持率保持高位。

技术委员会：按照每队一票的方式组成，每个团队必须在 Polkadot 或者Kusama 网络上具有成功且独立地执行过或被正式指定过协议实施的经历。理事会可以通过多数表决决定技术委员会成员的去留。

技术委员会是维护系统的最后一道防线。它的中心使命是检测系统中存在或即将出现的问题，例如代码中的错误或安全漏洞，以便快速提出解决方案并推进紧急公投。紧急提案需要同时至少四分之三的理事会成员和至少三分之二的技术委员会成员批准后才能提交。一旦提交，它就会快速进入全民投票，并与定期公投的时间表并行运行，投票周期更短，并且能够实现秒速颁布。这种情况下的批准机制与其他情况下的相同，即要么是简单多数，要么是在理事会一致批准的情况下的基于投票率偏好的批准模式。

我们强调，出于实际原因，技术委员会不是民主选举产生的，但相比之下，它的行动范围极小，并且没有单方面行动的权力，上述内容已经进行了解释。预计此机制足以用于非争议性错误修复和技术升级，但是考虑到要求的不断提高，类似政治敏感类或战略级紧急情况等情况下上述规则可能无效。

4.6.3 平行链卡槽配置

我们通过拍卖的方式来确保公链分配流程的公正、透明。从广义上讲，有兴趣获得平行链插槽的各方可参与以 DOT 计价方式的拍卖。出价最高的一方被宣布为获胜者，并被分配一个特定时间段使用的卡槽。同时，其参与竞标的Token 将被锁定，并在结束卡槽租赁结束时释放后。因此，该插槽的租赁成本对应于拥有该插槽而质押 Token 的机会成本。这种以 DOT 计价的质押也确定了平行链在 Polkadot 治理体系当中的投票权。

由于实施封标拍卖比较困难，且为了避免狙击出价，我们采用了具有追溯确定的结束时间特性的蜡烛拍卖[16]机制。详细来说，我们计划每隔几周进行一次拍卖，每次拍卖有四个连续的六个月时段以供租赁。参与方可以对一个、两个、三个或四个连续时段的任意组合进行出价，总共十个可能的时间组合方式，租赁期可以持续 6、12、18 或 24 个月。一旦拍卖开始，各方可以在固定的几个小时的窗口时间内，对这十个组合选项中的任何一个进行出价。允许一方提交多个出价，但需要满足以下条件：

a) 击败了相应时期的当前最高出价
b) 参与方没有成为两个或更多组合选项的临时获胜者或，且差距很大。

例如，（1.2）时段的获胜方是一个卡槽的前两个时段的获得者，此时它不能再对（4）这一时段出价。除非有其他人对前两个时段给出了更高出价，这时他才能对时段（4）出价。

此设计的既定目标是激励各方尽早投标并避免狙击出价，让资金较少的项目有机会赢得一个席位，从而确保 Polkadot 的去中心化属性。同时也是为了防止类似提高投标价格却无意中标的恶意攻击。

4.6.4 国库

国库会不断筹集资金。这些资金用于支付开发人员进行软件更新、用于实施由民主公投决定的任何更改、用于调整参数、以及维护系统的平稳运行。资金还可用于其他活动，例如营销活动、社区活动和推广活动。这最终由所有 DOT 持有者通过治理来控制，真正决定国库使用走向的是社区及其集体的想象力和判断力。

国库资金通过以下两种方式获得：

1. 来自验证人通过挖矿获得的部分奖励
2. 来自验证人削减费用和部分网络交易费

第一种方法允许我们保持固定的通货膨胀率，同时将验证人奖励与质押水平挂钩（参见第 4.5.1 节）：在每个 era，预铸 Token 和验证人的奖励之间的差额会被分配给国库。我们也认为将所有的削减的一部分转交国库是方便的：跟踪发生大量削减的事件，系统可能需要额外的资金来进行软件更新开发或新基础设施用以解决现有问题，或者可能通过治理来决定偿还部分被削减的质押。因此，被削减的 DOTs 存在国库是有意义的，而不是销毁后又立刻铸造更多的 DoTs。

4.7 密码学

在 Polkadot 中，我们需要通过不同的密钥和密钥类型区分不同的权限和功能。我们粗略地将这些分类为用户交互的帐户密钥，和管理节点的会话密钥。会话密钥确定了在节点管理中，除了认证过程外，不会有操作员干预。

4.7.1 账户密钥

账户密钥有一个相关联的余额，其中部分可以被锁仓，以便在质押、资源租赁以及治理中发挥作用，包括等待几种时间类型的解锁期。由于这些角色施加的限制不同，且多个解锁期是同时运行的。所以，我们允许不同锁仓活动采用不同的锁仓时间。

我们鼓励大家积极参与所有这些角色，但首先参与这些角色不可避免地偶尔需要账户签名。同时，如果将账户密钥保存在不太方便的地方会更具有物理方面的相对优势，比如保存在保险箱，但同时这也使签名就变得非常不便。为了避免这样的摩擦，我们提出了以下解决方案。

为质押而锁定资金的账户称为 stash 账户。所有 stash 帐户都需要注册链上证书，将所有验证人运行和提名的权利委派给某些控制人账户。同时，也选定一些代理密钥进行治理投票。在这种状态下，控制人账户和代理账户可以分别在质押和治理功能中为 stash 账户签名，但不能进行资金交易。

目前，账户密钥方面，我们支持 ed25519 [6]和 Schnorrkel/sr25519[9]两种规格。两者都是使用 Ed25519 曲线的类似 Schnorr 的签名，因此两者的安全级别相近。对于需要硬件安全模块（HSM）支持或其他外币密钥管理方法的用户，

我们推荐使用 ed25519。另一方面， Schnorrkel/sr25519 提供更多区块链友好功能，如分层确定性密钥派生 (HDKD)和多重签名。

特别是，Schnorrkel/sr25519 采用 Mike Hamburg 的 Decaf[18, §7]的 Ristretto 压缩算法[21]，Ristretto 压缩算法提供 Ed25519 曲线的 2 个无扭点作为质数群。避免此类辅引子表明 Ristretto 使在实现复杂化协议时更加安全。Polkadot 中大部分常规哈希都采用 Blake2b 算法，但是 Schnorrkel/sr25519 本身使用 STROBE 128 [19]，是基于 Keccak-f(1600)并提供非常适用于签名和非交互式零知识证明（NIZK）的哈希接口。

4.7.2 会话密钥

每个会话密钥在共识或安全方面扮演一个特定的角色。通常，会话密钥获得仅来自会话许可的授权，由代表一定质押量的某些控制人密钥签字。

任何时候，控制人密钥都可以暂停或撤销此会话许可，以及可以选择是否更换新的会话密钥。所有新的会话密钥都可以提前注册，而且大多数必须提前注册，便于验证人通过发放只在未来某个会话之后有效的会话许可，使其干净利落地过渡到新硬件。我们建议使用暂停机制进行紧急维护，并在会话密钥可能被泄露时使用撤销机制。

我们更倾向于会话密钥能够绑定到一台物理设备，以最大程度地减少意外风险的发生。我们要求验证器和操作员通过 RPC 协议发起会话许可，而不是通过会话密钥本身去操作。

几乎所有早期的 POS 网络在公钥基础设施上都有疏忽，间接的鼓励了跨设备复制会话密钥，从而降低了安全性并导致无意义的罚款。

我们对具体的组件或其关联会话密钥类型[6]并不进行在加密密码学方面的预先限制。

在 BABE 4.3.1 中，验证人使用 Schnorrkel/sr25519 密钥作为常规 Schnorr 签名，也可以用于基于 NSEC5[28]的可验证随机函数(VRF)。[3]

VRF 是伪随机函数(PRF)的公钥模拟，又名具有可分辨密钥功能的加密哈希函数，例如许多 MAC。当区块生产者 VRF 输出 VRFsk (r_e‖slot number)的得分足够低，那么，任何拥有 VRF 公钥的人可以验证区块是否在正确的 slot 中产生，但只有区块生产者才能通过他们的 VRF 密钥提前知道他们的插槽。

如[15]描述，我们为 VRF 输入提供一系列随机性 r_e，通过哈希所有 VRF 输出形成前一个会话，这就要求 BABE 密钥至少在使用之前的两个完整 epoch 完成注册。

我们通过对输入端签名者的公钥进行哈希以降低 VRF 输出的延展性，与 HDKD 一起使用时，可显着提高安全性。当输出端在其他地方使用是，我们对 VRF 的输入和输出信息都进行哈希，提高了其作为安全证明随机预言机使用时的可组合性。相关 Theorem 2 2Hash-DH 构造内容，请参阅第 32 页附件 C [15]。

在 GRANDPA 4.3.2 中，验证人应使用 BLS 签名进行投票，支持方便签名聚合并选择 ZCash BLS12-381 曲线以获得性能。这里存在的风险是由于数阈筛选

6 我们在 Polkadot 原生代码上总是执行加密，其原因不仅是因为“runtime”受到 WASM 性能损失的影响，更因为 Polkadot 的部分共识协议在“runtime”之外的 Substrate 模块中执行。

法的进步，BLS12-381 的安全性可能会大大低于 128 位。如果发生这种情况，我们预计将 GRANDPA 升级到另一条曲线来解决问题。

我们也将 libp2p 的传输密钥大致视为会话密钥，但它们不单单是给验证人本身使用，也包括传输哨兵节点的密钥。因此，操作者需要更多和它产生交互。

4.8 网络

在前面的部分中，我们讨论了节点将数据发送到另一个节点或其他节点组合，但没有具体说明这是如何实现的。我们这样做是为了简化模型并能够清楚地界定不同层之间的核心关注点。

当然，在真正的去中心化系统世界中，网络部分也必须是去中心化的——虽然网络上运行的高级协议是去中心化的，但是如果所有的通信都通过几个中央服务器，这显然不是好方案。举一个具体的例子：在某个安全模型中，包括传统的拜占庭容错设置，节点的建模可能是恶意的，但是缺少相关恶意的具体评估定义。一个安全模型要求必需有>1/3 的节点是诚实的。即当诚实节点>1/3 的时候，所有节点都可以随时实时地的相互信任的交流。然而，如果边界实际上被恶意的 ISP 控制，那么模型下的任何分析都会认为其相应的节点是恶意的。更重要的是，即使实际上中心化力量对众多节点没有任意执行权，但是如果底层通信网络是中心化的，就赋予了中心化的各方有能力破坏模型中>1/3 的节点，并进一步破坏这个网络的安全假设。

在本节中，我们概述并列举了我们在 Polkadot 中需要的通信原语，并勾勒出关于我们如何以去中心化的方式实现这些高级设计，同时随着我们不断对量产系统的推进，更多的细节将得到完善。

4.8.1 网络概况

如上所述，Polkadot 由一个独特的中继链与许多不同的平行链组成，同时中继链为平行链提供安全服务。为此，需要满足以下网络级别功能，大致分为以下方面：

1. 与所有区块链协议一样，中继链需要满足以下功能要求：
    a) 接受并分发来自用户的交易和其他外部的数据（统称为外部数据或外部源）
    b) 分发收集子协议的工件 4.2
    c) 分发终态子协议的工件 4.3.2
    d) 同步先前完成的状态

作为一个重要的示例，平行链可以根据上述结构自行选择执行与否，甚至可能重新使用相同的子协议。波卡的部分实现被构建成了一个单独的程序库，称为“Substrate”，专门进行基础框架构建。

2. 关于中继链和平行链交互，需要：
    a) 接受来自平行链收集人产生的平行链区块
    b) 分发包含有效性证明的平行链区块元数据
    c) 分发平行链区块数据并使其一段时间内可用(见 4.4.2)，以完成审计

3. 关于平行链间的交互，需要：
   a) 在平行链之间分发消息(见 4.4.3)，特别是从相关发件人到相关收件人的消息(见 4.4.3)。

对于上述每个功能要求，我们通过以下方式进行满足：

- 1(b), 1(c), 2(b) – 工件通过 Gossip 进行原样广播（即无需进一步编码）
- 1(a), 1(d), 2(a) - 实际上，一组节点向客户端提供相同的分布式服务。为了接受外部信息或区块，客户端将它们直接发送到服务节点；为了同步，客户端直接从服务节点接收可验证的数据。
- 2(c) - 特殊情况，如下。简而言之，数据将被擦除编码，以便不同的接收者接收到一个小的数据大小/接收到一小部分数据；碎片信息直接接通过 QUIC 发送。
- 3(a) - 特殊情况，如下。简而言之，消息直接通过 QUIC 发送；在这个过程中，发件箱被重新组合成收件箱，随后又可以批量传输给收件人。

我们将在接下来的几节中更详细地介绍这些内容。最后，再分享一下支持所有子协议的底层技术，即身份验证、传输和发现相关内容。

4.8.2 Gossiping

该子协议用于大多数中继链工件，通过此展示每个人或多或少需要看到相同的公开信息。它的部分结构也用于当节点离线一长段时间后需要同步以前未接收的任何新数据。

Polkadot 中继链网络在物理通信网络之上形成 Gossip 覆盖网络，作为去中心化广播媒体的有效方式。网络由已知数量的通过质押而受信任的节点（验证人），和来自未经许可的开放网络中的未知数量的不受信任节点（不执行验证的全节点）组成。（备注：一些不受信任的节点可能承担其他角色，例如平行链收集人、钓鱼人等）

目前实现了一种简单的基于推送型的方法，采用基于哈希的跟踪器缓存，以避免向对等节点发送重复信息，并采取一些限制，以避免最常见的垃圾邮件攻击：

- 工件只能按依存顺序接收；不允许对等节点乱序发送。尽管这会降低网络级的效率，但实施起来相对简单并能够提供可靠的安全性。
- 为有效地与发送方进行通信，明确发送方可以发送的信息顺序，peers 会定期根据链的最新信息更新对方。同时，验证人和收集人节点定期对中继链的最新头块信息进行更新通信。

关于使用 Gossip 协议的各类高级子协议的工件还需要满足更多具体的约束规则，以避免广播延误或进行不必要的工件。例如，对于 GRANDPA，对于各类型的投票，我们只允许每种类型的投票、轮数和选举人，只能获得两票，其他追加投票将被忽略。并且，仅允许有效验证的区块生产者对每一轮生产一个区块，其他追加的区块生产都将被忽略。

作为对哨兵节点的基本支持，本质上只有代理服务器与私人服务器邻近，进行最重要的安全相关的运行，如验证人角色。

网络拓扑是目前的一个薄弱点；节点通过执行随机查找的方式来实现以 ad-hoc 为基础的方式相互连接，通过地址簿执行随机查找。进一步的工作将沿着以下两个方向展开：

1. 受信任的节点将保留其部分带宽和连接资源，形成一个结构化叠加层，即具有确定性，又满足根据不同 era 转换的不可预测性。对于在哨兵背后运行的节点，通过代表这些节点的哨兵节点参与这个拓扑结构。
2. 对于剩余的可信节点的资源容量，以及对于整个不可信节点的资源容量，将通过基于延迟测量的方案进行组合，细节有待进一步商榷。值得注意的是，为了获得良好的安全属性，我们需要方案的设计不是简单基于“最近优先”，也会对远处节点进行组合。

从某种意义上说，这可以看作是受信任节点和其周围的不受信任节点形成了一个核心。但需要注意的是，受信任的节点会使用它们的一些资源来服务不受信任的节点。选择这两种拓扑形式有利于来缓解日蚀攻击，以及公链不受信任下的女巫攻击。

我们还在协调协议设置方面展开进一步工作，进一步减少由于许多发件人尝试将同一内容同时发送给同一收件人而造成的冗余；此外，也会考虑在保持安全性的同时取消排序限制。

4.8.3 分发服务

当 Polkadot 的某些部分向某个外部实体提供服务时，也将使用此子协议。即 1(a)接受中继链的交易，1(d)同步中继链的状态，以及 2(a)对接受上述列表中被校对过的区块。

在最初的实现中，仅在地址簿中查找一个特定的目标集，从这个目标集中选取几个节点，并连接到他们。对于 1(a)和 1(d)目标集是整个验证人集合，对于 2(a)目标集是平行链验证人集合，它们进行平行链客户端的校对。这两个都可以直接进行链上状态检索，实际上对于 1(a)来说，这与加入 gossip 网络的过程相同。

我们将在接下来的工作中进一步考虑传输层连接整个目标集的负载均衡问题，以及确保可用性。这可能需要增加地址簿的复杂性。

4.8.4 存储和可用性

该子协议解决章节 4.4.2 中描述的网络可用性和有效性相关话题。

考虑到可扩展性，Polkadot 不要求每个人对整体系统的状态进行存储，准确的说，即不需要所有区块都存储网络的全部状态。相反，每个平行链区块通过擦除码被分成几个碎片，这样每个验证人都有一个碎片，总共有 N 个碎片。出于安全因素考虑，擦除阈值设置为 ceil(N/3)。所有碎片信息由指定的收集人提交。且最初都可以在相关的平行链验证人处获得，（在这个角色中，平行链验证人也被称为初检员）然后这些分片信息将根据以下步骤进行分发：

1. 分发 - 每个平行链验证人最初都想要其中一个碎片，而平行链验证人也必须按一人一分片的要求进行分发；

2. 检索 - 审批检查者（即中继链验证人）需要确认有 ceil(N/3) 的验证人有他们的碎片，并且其中部分中继链验证人将尝试进行碎片检索；
3. 进一步检索 -作为可选项，其他非验证方也可能想要执行进一步检查，例如响应钓鱼人警报，再次提出 ceil(N/3)分片的需求。

这个子协议的目的是确保 ceil(N/3)这个阈值可行，以及可以在合理的时间内得到相关验证人中检索，直到至少完成了后面几个阶段的时候。我们将遵循一个类似 bittorrent 的协议，但有以下区别。

•对于分发和检索，接收者的集合是已知的。因此，除了 bittorrent 的 pull 语义之外，碎片可以通过已经拥有碎片的验证人提前推送；

•哨兵节点背后的验证人将使用这些作为节点作为代理，而不是直接发送；

•与中心化的追踪器不同,像谁拥有什么碎片信息这样的追踪,会通过中继链的 Gossip 网络进行传播。

对于初检员的预期是可以完全或者大部分连接；这也是校对协议的前提要求。审批员也应该完全或大部分连接，以帮助检索过程更快地完成。

除此之外，节点可以按照协议认为合适的方式与任何其他节点进行通信，类似 bittorrent。为了防止 DoS 攻击，他们应该实施类似 bittorrent 的资源限制，此外节点应该相互验证并且只与其他验证人通信，包括初检员和审批员。后期可选阶段的非验证方会被授予身份验证令牌以达到可以通信的目的。关于负载均衡议题，例如节点在随机选择中如何避免对其他节点的意外覆盖，我们在单独文件中进行详细讨论。

我们认为此部分组建不适合使用结构化叠加拓扑学的原因有以下几点：

1. 每个碎片是发送给特定的人员，而不是每一个人
2. (a)想要获得特定数据碎片的人，需要知道从哪里获取 – 例如从验证人，初检员处获得
   (b)其他想要随机非特定数据的人 – 例如审查员，希望任意 1/3 的碎片都可以重构

叠加拓扑学的应用场景要求刚好与上述情况完全相反：

1. 各数据碎片几乎会发给全员
2. 或者，人们需要特定的数据碎片，但不需要知道从哪里获取

例如，bittorrent 有类似的要求要求类似但没有采用结构化叠加，它通过需求偏差进行点对点的连接。

4.8.5 跨链消息

本节内容对 XCMP 消息传递子协议（章节 4.4.3）的相关网络设计进行分析。

简单回顾章节 4.4.3 内容，平行链之间能够相互发送消息。输出信息内容作为平行链 PoV 区块的一部分，由发出信息的平行链的收集人发送并递交给相应平行链验证人。并作为可用性协议的一部分分发给其他验证人。中继区块包含各平行链输入信息相对应的输出信息的元数据。因此，XCMP 网络的工作就是针对每个接受信息的平行链从其他发件箱中获取其输入消息。

值得一提的是，这一过程可通过 A&V 协议的擦除码片段检索完成，并不会增加额外复杂性，例如，通过 Gossip 网络，解码所有潜在发送方的发件箱。但

在初期——广播媒介用于两大方面：一、对接收方平行链播报感兴趣的数据进行广播；二、除了对接收方进行播报，也用于对发送给平行链的正在进行检索的数据进行播报。这是非常低效的，所以这被用作在通讯量较低情景下的早期网络初步实现方案。

进一步工作的展开从以下几方面展开，问题之一是如何有效地将所有发件人的发件箱转换为所有收件人的收件箱。这个问题一旦解决，任何收件人可以通过检索收件箱信息来定位以完成信息转换。我们注意到我们的 A&V 网络结构具有非常相似的通信要求 – 也就是，每个平行链区块的分片必须分发给其他每个验证人，反之亦然，每个验证人都必须接收每个平行链区块的分片。因此，我们的工作重心将放在 A&V 网络协议的扩展能力，以支持 XCMP 发件箱变成收件箱的转换能力。

另一个需要提及的重要区别是 A&V 中的分片存在的内置冗余，相比之下 XCMP 消息没有内置冗余，且所有信息必须进行全部有效分发。采用擦除码技术也是一个简单明了的解决方案，同时团队也在探索替代方案。

4.8.6 哨兵节点

有时，网络运营商出于运营安全的考虑，希望对其物理网络的各个方面进行安排。其中一些安排是独立的并且与任何去中心化协议的设计兼容，这些协议通常在上层拓扑结构中工作。然而其他一些配置需要特殊处理分布式协议，特别是影响节点的可获得性的相关配置。

对于此类应用场景，Polkadot 支持将全节点作为另一个全节点的哨兵节点运行，且这个全节点只能由这些哨兵节点访问。当为单个私有全节点运行多个哨兵节点时，这个方式最有效方式。简而言之，在协议方面，哨兵节点和其私有节点邻近，并通过结合额外元数据让其他节点和其私有节点沟通。在直接发送模式下，哨兵节点类似 TURN 服务器，不存在任何资源瓶颈限制，因为每个哨兵节点只服务一个私有节点。这些附加内容相当简单，更多细节可从其他地方获得。

如果您认为上述安全性优势不值得追加潜在成本，也可以不运行哨兵节点。

下面简要讨论基于这种方法下的安全权衡。受限制的物理拓扑的优势之一是，能够在跨多网关代理的情况下，提供负载平衡支持和 DoS 保护支持。如果软件中存在漏洞，间接还可以帮助保护私有节点 - 但请注意，这并不包括最严重的漏洞，比如一些漏洞在哨兵节点上提供任意执行权限，然后将其作为攻击私有节点的发射台。因此，即使当一个公共地址的服务代码写的很好的时候，我们也不认为它本身具有安全性保障，但哨兵节点可以帮助缓解这些情况，提供更多的安全性保障。

（另一种可能性是网络运营者运行较低级别的代理，例如 IP 或 TCP 代理，作为私有节点。这方面没有 Polkadot 协议的支持也可以完成。对比以上方案，哨兵节点的一个优势是来自作为 Polkadot 协议的一部分，通过哨兵节点的通讯已经进行了某种程度的验证和净化，较低级别的代理在这方面存在缺失。当然这里也可能存在漏洞，但这些被优先处理，因为它们是对抗整个网络的可用放大矢量）

4.8.7 授权、传输和发现

一般的安全协议与 Polkadot 类似，实体通过加密公钥相互关联。弱引用无法提供强安全性关联，例如，IP 地址，因为它们通常不是由实体本身控制，而是由它们的通信提供商控制。

然而，为了实现交流，我们需要在实体和其地址之间建立关联。Polkadot 使用与许多其他区块链类似的方案，即使用广泛使用的分布式哈希表（DHT），Kademlia[22]。Kademlia 是 DHT 的一种，采用 XOR 距离度量，常用于高流失网络。我们采用 Protocol Labs 的 libp2p Kademlia 并在执行过程中进行改进来达到使实体和其地址建立关联的目标。为防止日蚀攻击[20]，我们允许路由表足够大以包含最小限度的诚实节点，并且实现多路径路由的 S-Kademia 执行。

目前，地址簿服务也被用作主要发现机制——节点在加入网络时进行密钥/键(key)空间随机查找，并连接到任何一组返回的地址。同样，节点接受任何传入连接。这便于支持轻量化客户端和其他非特权用户的支持，但也容易造成 DoS 攻击。

进一步的工作将把发现机制与地址簿进行分离，如 gossiping 章节介绍到，打造更为安全的网络拓扑，还需要当前受信验证人集合授权部分传输级别的连接。然而，我们还需要保留接受来自未经授权实体的输入连接的能力，这需要在资源基础上进行限制，与受信实体之间保持平衡。

进一步的工作也会将地址簿的实现与其接口解耦，因此，例如，我们可以将一部分放在链上。这与基于 Kademlia 的地址簿有不同的安全权衡，其中一些不在 Polkadot 的当前工作范围内，例如位置隐私。通过提供不同的可能性，我们希望通过多样的节点组合来满足不同的安全要求。

5 远期计划

在未来工作计划中，我们将专注于 Polkadot 的一系列扩展工作。我们希望添加激励钓鱼人的模型，以确保他们有足够的激励来报告恶意行为。关于免信任消息传递，我们正在进行 SPREE A.1 的开发以实现信息去信任传递。我们也将致力于进一步提高 Polkadot 的可扩展性，例如嵌套式中继链的研究。此外，桥接协议 A.2 也是我们工作计划的一部分，例如进行与比特币、以太坊和 Zcash 的桥接。此外，为了提高可用性，我们计划启用平行线程，具有与平行链相同的功能，但是租赁时间更短，以及有不同的收费模式。

致谢

我们要感谢来自 Web3 基金会的 Bill Laboon 对本文的反馈，以及 Parity Technologies 开发人员的有益建议和帮助讨论。

参考文献

[1]. Availability and validity scheme.https://research.web3.foundation/en/latest/polkadot/Availability_and_Validity/.

[2]. Blind assignment for blockchain extension(babe).https://research.web3.foundation/en/latest/polkadot/BABE/Babe/.

[3]. Visa inc.at a glance,Dec 2015.Accessed:2020-02-25.

[4]. Mustafa Al-Bassam,Alberto Sonnino,and Vitalik Buterin.Fraud and data availability proofs:Maximising light client security and scaling blockchains with dishonest majorities.arXiv preprint arXiv:1809.09044, 2018.

[5]. Handan Kılın¸c Alper.Consensus on clock in universally composable timing model.Cryptology ePrint Archive,Report 2019/1348, 2019.https://eprint.iacr.org/2019/1348.

[6]. Daniel Bernstein, Niels Duif, Tanja Lange, Peter Schwabe, and Bo-Yin Yang.High-speed high-security signatures.Journal of Cryptographic Engineering volume,2012(2):77-89, 2012.

[7]. Markus Brill, Rupert Freeman, Svante Janson, and Martin Lackner. Phragm´ ens voting methods and justified representation.In Thirty-First AAAI Conference on Artificial Intelligence,2017.

[8]. Ethan Buchman, Jae Kwon, and Zarko Milosevic.The latest gossip on bft consensus.arXiv preprint arXiv:1807.04938,2018.

[9]. Jeffrey Burdges.schnorrkel:Schnorr vrfs and signatures on the ristretto group. https://github.com/w3f/schnorrkel,2019.

[10]. Vitalik Buterin.Ethereum:A next-generation smart contract and decentralized application platform,2014.Accessed:2016-08-22.

[11]. Vitalik Buterin and Virgil Griffith.Casper the friendly finality gadget.arXiv preprint arXiv:1710.09437,2017.

[12]. Alfonso Cevallos and Alistair Stewart.Validator election in nominated proof-of-stake.arXiv preprint arXiv:2004.12990,2020.

[13]. Tarun Chitra.Competitive equilibria between staking and on-chain lending.arXiv preprint arXiv:2001.00919,2019.

[14]. Kyle Croman, Christian Decker, Ittay Eyal, Adem Efe Gencer, Ari Juels, Ahmed Kosba, Andrew Miller, Prateek Saxena, Elaine Shi, Emin Sirer, Dawn Song, and Roger Wattenhofer.On scaling decentralized blockchains.volume 9604, pages 106-125,02 2016.

[15]. Bernardo David, Peter Gaˇ zi, Aggelos Kiayias, and Alexander Russell.Ourobor-os Praos: Anadaptively-secure, semi-synchronous proof-of-stake blockchain. In Annual International Conference on the Theory and Applications of Cryptographic Techniques, pages 66-98.Springer,2018.https://eprint.iacr.org/2017/573.

[16]. Sascha Fllbrunn and Abdolkarim Sadrieh. Sudden Termination AuctionsAn Experimental Study.Journal of Economics & Management Strategy,21(2):519-540, June 2012.

[17]. Juan Garay, Aggelos Kiayias, and Nikos Leonardos.The bitcoin backbone protocol: Analysis and applications. In Annual International Conference on the Theory and Applications of Cryptographic Techniques, pages 281-310. Springer, 2015.

[18]. Mike Hamburg. Decaf: Eliminating cofactors through point compression. In Rosario Gennaro and Matthew Robshaw, editors, Advances in Cryptology-CRYPTO 2015, pages 705-723, Berlin, Heidelberg, 2015. Springer Berlin Heidelberg. https://eprint.iacr.org/2015/673.

[19]. Mike Hamburg.The STROBE protocol framework.IACR ePrint 2017/003, 2017. https://eprint.iacr.org/2017/003 and https://strobe.sourceforge.io.

[20]. Ethan Heilman, Alison Kendler, Aviv Zohar, and Sharon Goldberg.Eclipse attacks on bitcoin's peer-to-peer network. In 24th USENIX Security Symposium (USENIX Security 15),pages 129–144, Washington, D.C., August 2015. USENIX Association.

[21]. Isis Lovecruft and Henry de Valence. Ristretto.https://ristretto.group. Accessed: 2019.

[22]. Petar Maymounkov and David Mazi` eres. Kademlia: A peer-to-peer information system based on the xor metric.In Revised Papers from the First International Workshop on Peer-to-Peer Systems, IPTPS '01, pages 53-65, London, UK, UK, 2002. Springer-Verlag.

[23]. Silvio Micali. ALGORAND: the efficient and democratic ledger.CoRR,abs/1607.01341, 2016.

[24]. Silvio Micali, Michael Rabin, and Salil Vadhan.Verifiable random functions. In 40th Annual Symposium on Foundations of Computer Science (Cat.No. 99CB37039), pages 120-130.IEEE, 1999.

[25]. David Mills et al.Network time protocol.Technical report, RFC 958, M/A-COM Linkabit,1985.

[26]. Satoshi Nakamoto.Bitcoin: A peer-to-peer electronic cash system, Dec 2008. Accessed:2015-07-01.

[27]. Ryuya Nakamura, Takayuki Jimba, and Dominik Harz.Refinement and verification of cbc casper. Cryptology ePrint Archive, Report 2019/415, 2019. https://eprint.iacr.org/2019/415.

[28]. Dimitrios Papadopoulos, Duane Wessels, Shumon Huque, Moni Naor, Jan Vˇ celák, Leonid Reyzin, and Sharon Goldberg. Making NSEC5 practical for dnssec. IACR ePrint Report 2017/099, 2017. https://eprint.iacr.org/2017/099.

[29]. Luis Sánchez-Fernández, Edith Elkind, Martin Lackner, Norberto Fernández, Jesús A Fisteus, Pablo Basanta Val, and Piotr Skowron.Proportional justified representation. In Thirty-First AAAI Conference on Artificial Intelligence, 2017.

[30]. Luis Sánchez-Fernández, Norberto Fernández, Jesús A Fisteus, and Markus Brill. The maximin support method: An extension of the d'hondt method to approval-based multiwinner elections. arXiv preprint arXiv:1609.05370, 2016.

[31]. Elaine Shi. Analysis of deterministic longest-chain protocols. In 2019 IEEE 32nd Computer Security Foundations Symposium (CSF), pages 122-12213. IEEE, 2019.

[32]. Alistair Stewart. Byzantine finality gadgets. Technical Report, 2018. https://github.com/w3f/consensus/blob/master/pdf/grandpa.pdf.

[33]. Gavin Wood. Polkadot: Vision for a heterogeneous multi-chain framework. White Paper,2016.

[34]. Vlad Zamfir. Casper the friendly ghost: A correct-by-construction blockchain consensus protocol.2017.

[35]. Alexei Zamyatin, Dominik Harz, Joshua Lind, Panayiotis Panayiotou, Arthur Gervais, and William J. Knottenbelt. XCLAIM: trustless, interoperable, cryptocurrency-backed assets.In 2019 IEEE Symposium on Security and Privacy, SP 2019, San Francisco, CA, USA, May 19-23, 2019, pages 193-210. IEEE, 2019.

附录 A

A.1 SPREE

SPREE（Shared Protected Runtime Execution Enclaves）是一种让平行链拥有共享代码的方式，并且沙盒化代码的执行和状态。从平行链 A 的角度来看，它能在多大程度上信任平行链 B？Polkadot 的共享安全保证了 B 的代码的正确执行，且保证对 B 代码的执行安全性与执行 A 代码的安全性的一样高。然而，如果我们不知道 B 代码本身（即使知道），B 的治理机制也可以改变其代码，而我们不信任这个改变。如果要改变这种情况，我们就需要知道 B 的部分代码，且对这部分代码不会被 B 的治理机制左右，那么这部分代码可以向 A 发送消息，如此便知 B 代码的正确执行会对这些消息产生怎样的作用，所以共享安全给我们提供了我们需要的保证。

SPREE 模块是放置在中继链中的一段代码，平行链可以选择加入。这段代码是该链状态转换验证功能（STVF）的一部分。SPREE 模块的执行和状态被沙盒化，与 STVF 的其他执行分离。远程链上的 SPREE 模块可以被 XCMP 寻址。平行链能收到的信息由其 SPREE 模块决定（这对于想要使用任何 SPREE 模块的链来说，这是一个内生强制执行）。

我们期望 XCMP 发送的大多数消息将从一个链上的 SPREE 模块发送到另一个链上的同一 SPREE 模块。当 SPREE 模块被升级时，涉及到将更新的代码上传到中继链，并安排一个更新的区块号，它将在所有平行链的下一个区块上进行更新。这样做是为了保证一个版本的 SPREE 模块在一个链上向另一个链上的相同模块发送的消息时，消息永远不被过去的版本 SPREE 收到。因此这些被传送的消息的格式，既不需要向前兼容，也不需要取参考其它标准。

举个例子，从 SPREE 得到的安全保证，如果 A 有一个原生 A Token，我们希望确保平行链 B 不能铸造这个 Token。这一点可以通过 A 在 A 的状态下为 B 保留一个账户来达到。但是，当 B 的账户想要发送一些 A Token 给第三个平行链 C 的时候，B 需要通知 A。A 的 SPREE 模块中负责 Token 的部分，允许 Token 在不需要核算的情况下进行转移。A 的模块只需向 B 的相同模块发送一个消息，告知将 Token 发送到某个账户。然后，B 可以 Token 发送给 C，C 也可以用同样的方式发送 Token 给 A。模块本身将对 B 链上账户中的 Token 进行核算，Polkadot 的共享安全以及模块的代码将强制执行 B 永远不能铸造 A 的 Token。XCMP 保证信息将被传递，SPREE 保证信息将被正确编译执行，这意味着每次传输只需发送一个信息，并且是去需信任的。这一点的应用不仅试用于 Token 传输，也意味着信任最小化协议的设计会容易得多。

SPREE 的部分设计和实现还没有完全设计出来。目前这些归功于 reddit 用户 u/Tawaren 提出的 SPREE 的最初想法。

A.2 与外链的互操作性

Polkadot 将承载一些用于链接其他区块链的桥接组件。本节将重点讨论与 BTC 和 ETH（1.x）的桥接，因此将主要论述桥接基于 POW 共识的区块链方式。桥接设计受到 XClaim 的影响和启发，对于桥接逻辑有两个重要部分：一个是桥接中继，它将尽可能地解析桥接链的共识；另一个是银行，其中涉及拥有桥接链 Token 的权益参与者。桥接中继需要能够

对桥接链进行共识验证，并在桥接链上验证交易包含证明信息。一方面，银行可以被桥接链上的用户用来锁定 Token，作为他们想在 Polkadot 上获得相应资产的抵押，例如，PolkaETH 或 PolkaBTC；另一方面，用户可以使用银行将这些资产赎回到桥接链的 Token。桥式中继的目的是把桥接链的轻客户端逻辑尽可能多的放在桥式中继上，这是一种可行的 BTC-Relay 方案。然而，平行链上的加密和存储要比 ETH 智能合约中便宜的多。我们的目标是将桥接链的所有区块头和某些交易的包含证明放在桥接链的区块中。这足以决定一个交易是否在一个可能是最终的链上。比特币和 ETH1.0 的桥接中继的想法是有一个最长链的桥接链，冲突通过投票/证明方案来解决。

A.3 与其他多链系统的对比

A.3.1 ETH2.0

以太坊 2.0 承诺将逐步过渡到 PoS 共识协议，并部署分片来提高速度和吞吐量。Polkadot 和以太坊 2.0 的设计之间有广泛的相似之处，包括类似的区块生成和最终性确定装置。

以太坊 2.0 中的所有分片都是基于智能合约的同质链运行，而 Polkadot 中的平行链是独立的异构区块链，其中只有一些链支持不同的智能合约语言。乍一看，这简化了在 Ethereum 2.0 上的部署，但分片间的交互合约会使以太坊 2.0 的设计复杂化。我们有一个智能合约语言 Ink!，它的存在是为了让智能合约代码可以更容易地被迁移到成为平行链代码。我们认为，平行链这种固有的专注于自己的基础设施应该比智能合约更容易支持更高的性能。

以太坊 2.0 要求验证者的质押正好是 32ETH，而 Polkadot 固定了一个验证者的目标数量，并试图用 NPoS 最大化支持质押（见 4.1 节）。在理论层面上，我们认为 32ETH 的方法导致验证者比 NPoS 更不 "独立"，这削弱了整个协议的安全假设。然而，我们承认基尼系数在这里很重要，这使 Ethereum 2.0 在 "独立性 "方面具有初步优势。我们希望 NPoS 也能使 Dot 持有人更多的参与，将余额低于 32ETH。

以太坊 2.0 没有与 Polkadot 的可用性和有效性协议完全类似的协议（见第 4.4.2 节）。然而，我们确实从 Ethereum 提案[4]中得到了使用擦除码的想法，其目的是支持轻量级客户。

以太坊 2.0 中的验证者被分配到每个分片，用于证明分片的区块，作为 Polkadot 中的平行链验证者，从而构成分片的委员会。委员会成员从分片的完整节点收到随机选择的代码片段的 Merkle 证明，并验证它们。如果所有的片段都被验证，并且没有欺诈证明被宣布，那么该区块被认为是有效的。这个方案的安全性是基于在委员会中有一个诚实的多数，而 Polkadot 的方案的安全性是基于在平行链验证者或辅助检查者中至少有一个诚实的验证者（见 4.4.2 节）。因此，Ethereum 2.0 中的委员会规模与 Polkadot 中的平行链验证者规模相比要大得多。

以太坊 2.0 中的信标链与 Polkadot 的中继链一样，都是股权证明协议。同样，它有一个名为 Casper[11][34]的最终性确定装置，如同 Polkadot 的 GRANDPA。Casper 也像 GRANDPA 一样结合了最终确定性和拜占庭协议，但 GRANDPA 比 Casper 给出了更好的有效性属性[32]。

A.3.2 Sidechains

区块链扩容技术的另一种方式是侧链技术[7]。侧链解决方案同时也用于互操作性问题的解决，此类方案解决了侧链与主链之间的桥接问题。例如，Eth1.0 中引入了许多侧链，这些侧链有助于提高可扩展性，如 Plasma Cash 和 Loom[8]。其中，Cosmos 是极具代表性的侧链桥接解决方案，我们将在下节阐述 Cosmos[9] 和 Polkadot 的差异。

A.3.3 Cosmos

Cosmos 是一个旨在解决区块链互操作性问题的系统，这对提高去中心化网络的可扩展性至关重要。在这个意义上，这和 Polkadot 有表层相似之处，因此，Cosmos 包含类似 Polkadot 子组件的功能组件，例如，Cosmos Hub 用于 Comos 各区域之间信息传输，这与 Polkadot 中继链监督 Polkadot 各平行链之间的信息传递类似。

然而，这两个系统之间存在着重大差异。最重要的是，虽然 Polkadot 系统作为一个整体是一个分片的状态机（见第 4.2 节），而 Cosmos 的设计不包含对各区状态的整合，因此，各区的状态并不集中反映在 Hub 的状态中。基于此设计，Cosmos 没有在各区之间提供共享的安全性，这是与 Polkadot 的一大不同之处。这也导致 Cosmos 的跨链信息不再是无信任的。也就是说，为了对发送方的消息采取行动，接收区需要完全信任发送区。如果我们把 Cosmos 系统视为一个包含所有区域的整体来进行分析，那么，就像对波卡系统进行分析一样，系统的安全性应该能够保障最不安全的区域，同时，Polkadot 的安全承诺保证了经过验证的平行链数据可以在以后的时间里进行检索和审计（见 4.4.2 节）。然而，Cosmos 系统下用户需要相信区域运营商会保留链的历史状态。

值得一提的是，使用 SPREE 模块，使得 Polkadot 的安全性比共享安全更高一等。当一个平行链注册了一个 SPREE 模块，Polkadot 保证该平行链收到的部分 XCMP 消息已经由预定义的 SPREE 模块的代码集进行了处理。Cosmos 系统没有提供类似的跨区信任框架。

Cosmos 和 Polkadot 之间的另一个重要区别在于区块产出和最终完成方式的不同。Polkadot 中，由于所有的平行链状态都与中继链状态紧密相连，平行链可以与中继链暂时分叉。这允许区块产出与终态逻辑脱钩。因此，Polkadot 的区块可以在未最终完成的区块上生成，并且多个区块可以同时生成终态区块。另一方面，Comos Zone 依赖于 Hub 状态的即时确定性来执行跨链操作，因此，延迟确定会停止跨区域操作。

词汇表 B

名称	描述	符号	引用
BABE	一个随机分配当选验证人的机制，以生产某一区间段的区块		4.3.1
BABE Slot	一个生产中继链区块的时间单位，大约是 5 秒	sl	4.3.1
Collator	协助验证者进行区块生产。一组收集人被定义为$\mathcal{C}$	$\mathrm{c}(\mathcal{C})$	3.1
Dot	Polkadot 原生 token		4.5
Elected validators	一组获选验证人	$\mathcal{V}$	
Epoch	一个由 BABE 产生随机性的时间单位，大约 4 个小时	e	
Era	决定一个新的验证人组合的时间单位，大约 1 天		
Extrinsics	输入数据提供给中继链，用于转换状态		4.2
Fishermen	监视网络的不当行为		3.1
Gossiping	将每个新收到的消息广播给同伴		4.8.2
GRANDPA	敲定区块的机制		4.3.2
GRANDPA -Round	GRANDPA 算法的一种状态，引导区块终态确定		4.3.2
Nominator	质押利益相关方选举提名验证人。一组提名者被定义为 N	$\mathrm{n}(\mathcal{N})$	3.1
NPoS	Nominated Proof-of-Stake - Polkadot 的 PoS 版本。被提名的验证人被选出来，能够产生区块		4.1
Parachain	异质性独立链	P	
PJR	比例--合理--代表--确保验证人代表尽可能多的提名人的少数群体		4.1
PoV	有效性证明 - 验证者可以在没有完整状态的情况下验证一个区块的机制		4.4.1
Relay Chain	确保平行链间达成全球共识		4.2
Runtime	Wasm blob，其中包含状态转换函数功能，以及 Polkadot 要求的其他核心操作		4.2
Sentry nodes	专门的代理服务器，转发往来于验证人的流量		
Session	一个 session 是一个时间段，有一组恒定的的验证人，验证人只能在 session 变化时加入或退出验证人集合		
STVF	状态-转换-验证-功能--Runtime 的一个功能，用于验证 PoV		4.4.1
Validator	选举产生且作为最高权限方，有机会通过 BABE 被选中并且生产一个区块，一组候选验证人被定义为 C。选举选出的验证人被定义为 n_{val}	$\mathrm{v}(\mathcal{V})$	3.1
VRF	可验证随机函数(密码学函数),通过以上过程决定区块产出的获选验证者		4.3.1
XCMP	平行链互相发送消息采用的协议		4.4.3