链滴

比原链 BBFT 如何让共识更快——兼论 BBFT 与 FBFT/HotStuff 的比较

作者：bytom

原文链接：https://ld246.com/article/1562836765274

来源网站：链滴

前言

近日比原链(BYTOM)技术团队发布了Bystack区块链BaaS平台，其中包括侧链的共识算法BBFT(Bystak Byzantine Fault Tolerance)。笔者将在这篇文章中阐述比原链BBFT尝试解决的问题以及分析BBFT其他各家共识协议的主要差异。BBFT是一个PBFT的变形，它的原理与PBFT一脉相承。若想深刻理解BFT的巧思，则必须进入PBFT的脉络推敲。早在区块链藉由比特币的大红大紫之前，PBFT就作为共识议存在于世界上了。由Castro和Liskov于1999年发明，它是一个具有20年历史的经典设计，它的发是为了解决分布式系统中的一个经典问题：拜占庭将军问题。直到今日，PBFT仍蕴含许多值得反复推的巧思，不断启发后世发明出更好的协定。

PBFT基本的运作流程

PBFT是一个具有二轮投票的三阶段协议，每个视域(View)都会有一个特定的节点作为领导节点(Primar/Leader)，负责通知所有节点进入投票流程。各节点则会经历Pre-prepare/Prepare/Commit这三个段，并依据接收的讯息决定是否投票/进入下一阶段，每个节点投完票后将讯息发给所有其他的节点若个节点在两阶段投票之后取得多数共识，则各节点可以更新本机的状态，结束这一回合。 视域变换(iew-change)仅当多数节点发起时执行，当目前的领导节点并未正常执行任务时，这可以替换当前的导节点，保证协议正常运作。

PBFT的特性

PBFT与中本聪共识(区块链)有相当不同的特性：PBFT是一个许可制的、基于领导节点的、基于通讯的安全性重于活跃性的共识协定。

- 许可制的(Permissioned)：PBFT并非完全开放的，这是由于PBFT需要让节点能够验证彼此的讯息及精准掌握节点的数量，区块链则是属于对任何人都开放的非许可制(Permissionless)。
- 基于领导节点的(Leader-based)：也就是先决定领导节点(Leader)，再由领导节点送出提议，这样最直接的好处就是不需要浪费自己的运算资源去争取当领导节点的机会，然而缺点就是只有在视域变时才轮替领导节点，成为领导节点的机会并不公平，缺乏加入网络的诱因；区块链则是在多个提案中择工作证明难度最高的区块作为共识，虽然这样会造成运算资源的浪费，但是成为出块者的机率大致公平的，其与算力成正比。
- 基于通讯的(Communication-based)：PBFT的安全性奠基于3阶段投票，虽然不必如工作证明般耗大量计算资源，但数量庞大的通讯也造成可扩展性的瓶颈——就算是号称最实用的PBFT，也无法展到1000个以上个节点。不仅如此，PBFT使用消息验证代码(MAC)，每投一轮票就需要每一个节点证一次讯息，大量的签名/验证也是另一个潜在的瓶颈。
- 安全性重于活跃性的(Safety over Liveness)：PBFT不论在何种网络假设下(同步/异步)都能确保安性，几乎不可能出现分岔，因此具有实时敲定(Instant Finality)的特性；相对地，区块链则是活跃性于安全性，其安全性有赖于同步的网络，而具有复数个共识(及分岔)的情况也相当常见，需要经过一数量的区块「确认」才能保证其不再分岔的机率足够大。

PBFT的问题

首先，PBFT中的每个节点都需于每一轮投票中做n-n的通讯，假设n为1000，则每一次的共识都需要少100,000次的通讯，尽管PBFT已经是BFT家族当中最实用的协议，这么巨量的通讯需求仍是扩展的颈。

如何提升效率?

聚合签名

为了提升效率，一个直觉的思路是：避免n-n的通讯。我们可以指定网络中的某节点作为协调者来发送接收每个节点的投票，这样每个节点都只需要向协调者发送讯息即可，从而避免了n-n的通讯。然而在这样的情境下，协调者有作恶的可能，因为协调者可以在未确实接收到指定数量的讯息前便执行下轮投票或者进行状态更新。因此，我们可以使用门坎签章(Threshold Signature)来保证协调者的正当为，门坎签章的可以保证：需集合超过门坎数量(t-of-n)的签章才有效。也就是说，我们可以指定：有当协调者集合 2f+1 个门坎签章后，协调者才能带着合法的签名继续推进共识。Harmony FBFT便一个使用聚合签名以提升效率的BFT家族协议。

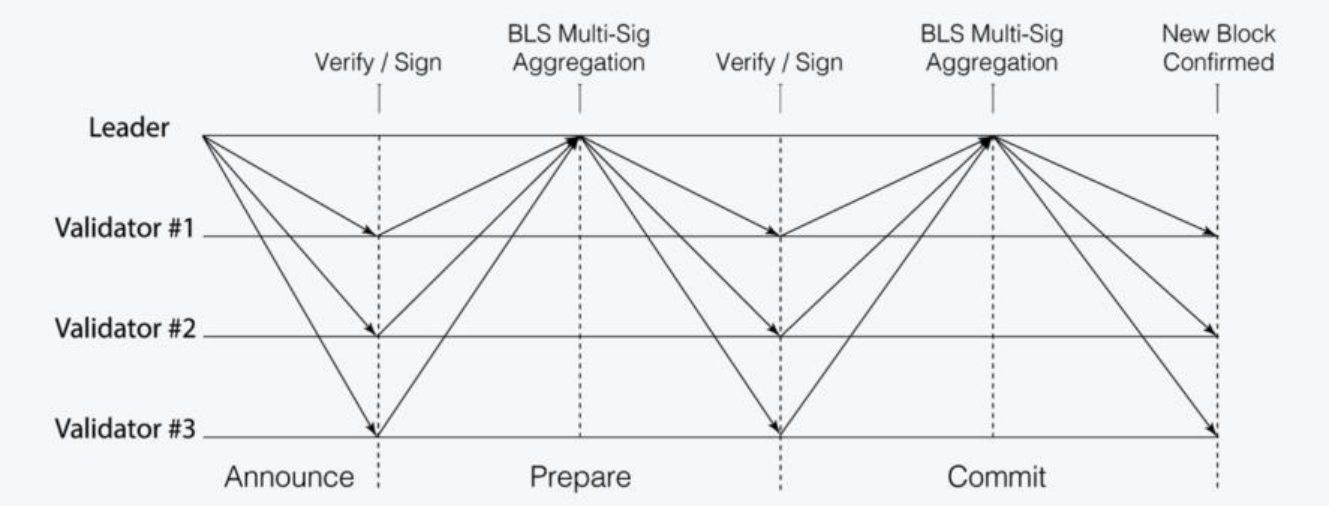

图1：FBFT Signature Aggregation

管线设计

每个内容都必须经过二轮投票/三个阶段才能达成共识，如果有m个内容就需要执行2m次投票。管线计(Pipelining)可以减少投票的次数，它的基本思路如下：让每个节点在投第 i 轮的prepare阶段时，时也是对其前一个内容 i-1 的commit阶段投票。这样做便可以节省对同一个内容重复投票的冗余，幅提升效率。这样的思路首见于2018年发表的HotStuff协议。

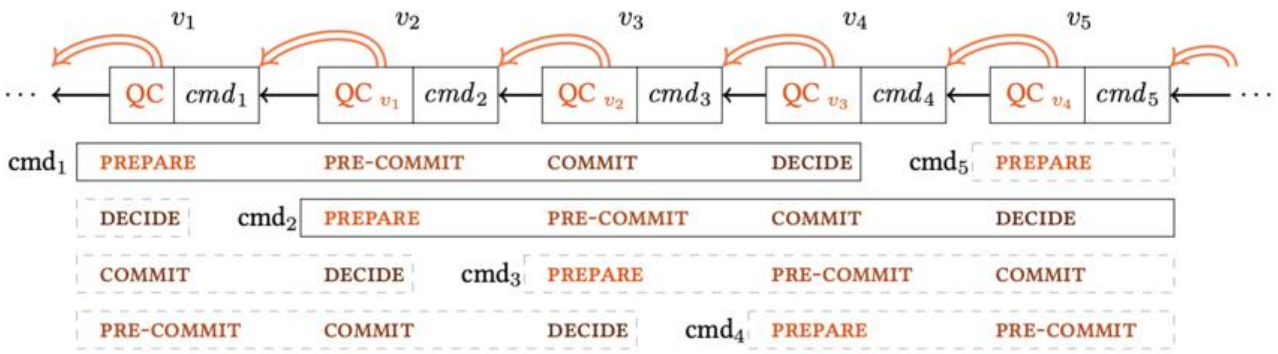

图2：HotStuff Pipelining

只让部分节点参与共识：最小生成树

另外一种提高效率的方法，就是避免使所有的节点参与共识，这也正是比原链BBFT采取的作法。在BBT中，节点分为三种：Consensus Node/Gateway Node/Leader Node，这些节点形成树的结构，为网络中节点的最小生成树(Minimal Spanning Tree)，可能由分布式算法得出，或是由外部服务提。树叶的节点即为Consensus Node；树根为Leader Node；其他部分为Gateway Node。每种节点有分别的任务：Consensus Node负责进行投票；Gateway Node则不需参与投票，但必须负责聚合Consensus Node送来的签章；Leader Node负责与其他Leader Node交换讯息。BBFT的运作流程下图所示，BBFT的共识过程，便是讯息由树根向树叶传播再回到树根的过程。

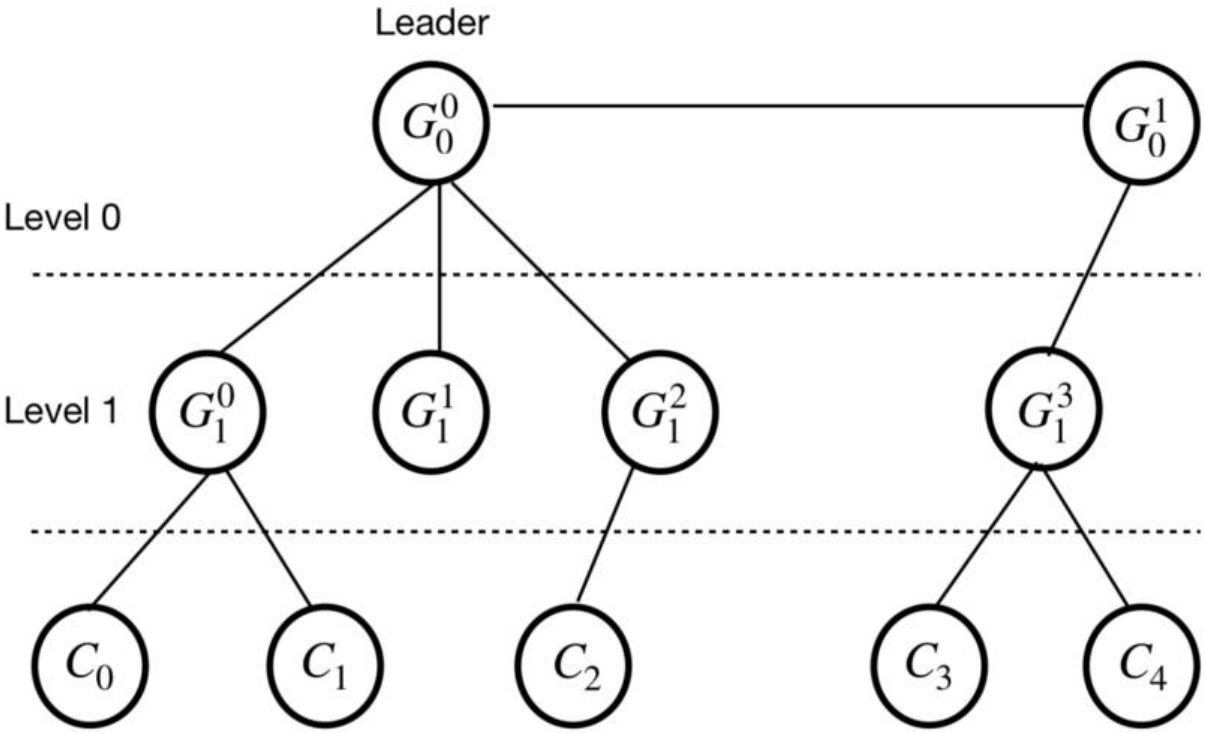

图3：BBFT: Minimal Spanning Tree

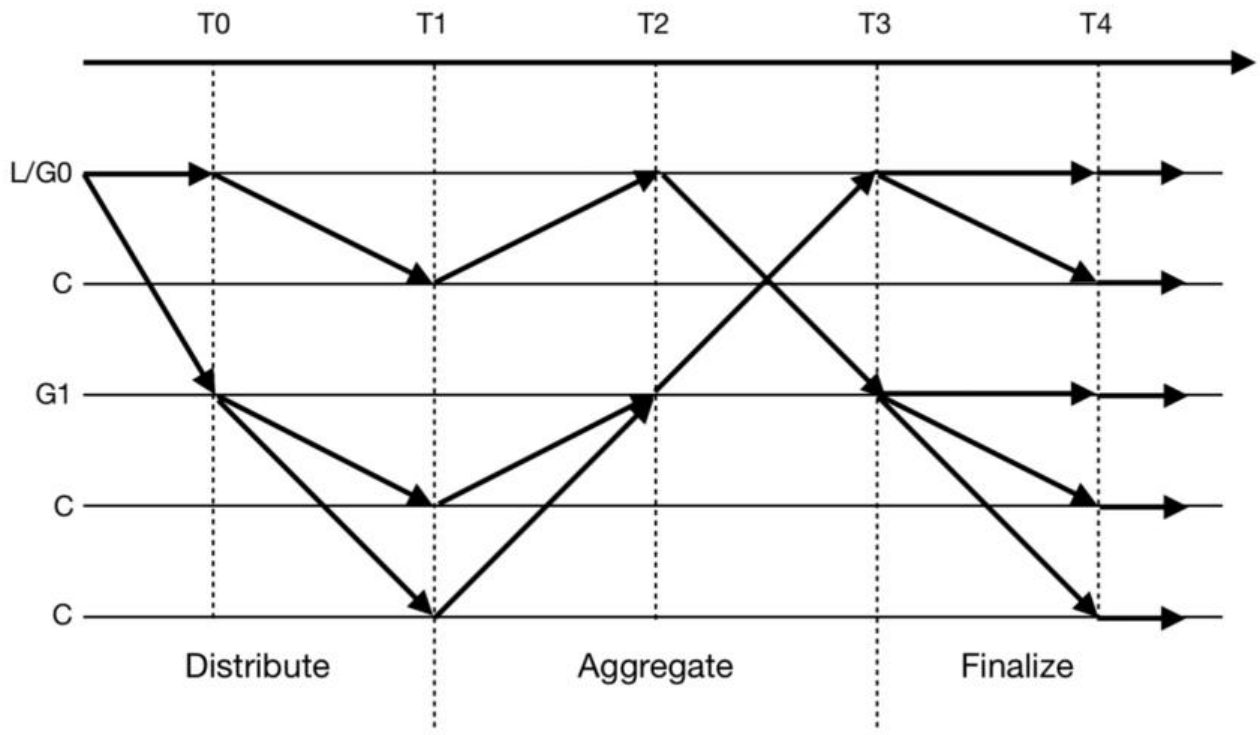

图4：BBFT Process

如何确保正确性(Safety and Liveness)?

在为PBFT带入新技术以提升效率的同时，也必须确保协议本身的安全性与活跃性。接下来我们来看看上述的协议是如何确保这两者。

视域变换(View-change)

FBFT沿用了PBFT的视域变换，即在正常情况下并不更换领导节点，仅有当超过2f+1个节点发起视域换才会更迭领导节点。视域变换虽然本身是一个能够替换作恶领导节点的机制，但它同时要求协议必具有3个阶段，才能保证协议的安全性(即不分岔)。

领导节点轮替(Rotating Leader)

HotStuff另一方面则引入了领导节点轮替的机制，在每个回合都更换领导节点，如此来回避视域变换额的通讯成本。领导节点轮替也常见于许多BFT家族的协议，算是目前保障安全性机制的主流。

混合式(Hybrid)

比原链BBFT则取各家所长，同时应用了视域变换与领导节点轮替，等于是上了双重保险。不过值得注的是，目前的BBFT技术白皮书仅有一轮投票的模型，并未提出两轮投票/三阶段共识的模型。另外，导节点轮替的顺序也将基于各节点的权益(Stake)，若节点出现违反协议的行为则该节点会遭受惩罚。

BBFT的挑战

综合以上的分析与比较，BBFT目前有几个显著的挑战。首先，是最小生成树的产生方式，如何同时兼去中心化与效率？其次是BBFT仅采取单轮投票作为共识，在引入视域变换的情况之下，可能会发生分，这样的网络也会遭受日蚀攻击的威胁。最后，引入门坎签章的前提下，势必需要引入分布式私钥生协议(Distributed Key Generation)以共同生成私钥，这部分于技术白皮书中尚未提及，却是可能造瓶颈的潜在因子。

结语

本篇文章简介了PBFT的特性及其效能问题，并比较了FBFT/HotStuff/BBFT等协议针对效能问题的解思路，最后归纳出比原链BBFT的未来挑战，希望能帮助读者更理解BBFT的精髓。

参考资料

- 比原链BBFT
- PBFT
- HotStuff
- Harmony FBFT

作者：Juin Chiu