Extended Abstract: GRANDPA Finality Gadget

Alistair Stewart

November 20, 2019

Abstract

We present GRANDPA, a finality gadget, that is a protocol that can be used to provide provable finality for a blockchain. It works in addition to a block production mechanism and a chain selection rule, that on their own would only provide eventual consensus. The design of GRANDPA aims at separating these two protocols as cleanly as possible and obtain formal guarantees for the finality gadget. GRANDPA attempts to finalise the prefix of the chain that 2/3 of voters agree on, whether that is one or thousands blocks. It has been implemented by Parity Technologies and deployed on large testnets for the Polkadot protocol. We also present properties GRANDPA achieves and review GRANDPA's advantages in flexibility over comparable protocols.

1 Introduction

We consider the question of finality for blockchain protocols: when will a block never be reverted? Many such protocols, such as the original blockchain, Bitcoin [7], have the property of eventual consensus - that an ever growing prefix of the chain will be agreed upon by all participants forever onward. But they generally only give probabilistic finality on a specific block - that under some assumptions about the network and participants, if we see a few blocks building on a given block, we can estimate the probability that it is final. But what we'd prefer is to have provable finality - for example a signed statement by some authorities, the set of whom can be tracked, that the block is final. This is useful to prove what happened to light clients, who do not have the full chain or are not actively listening to the network, and to communicate with other chains, possibly as part of a scalability solution, where not anyone receives or stores all the data in the system.

A popular family of consensus mechanisms for blockchains involves getting Byzantine agreement on each block [3, 5]. This gives provable finality immediately. However this is slow if we have a large set of participants in the Byzantine agreement. The approach that we will take is similar to the approach that Ethereum (https://www.ethereum.org) [10] plans to take with Casper the Friendly Finality Gadget (Casper FFG) [4], which combines eventual finality and Byzantine agreement. We will use a block production mechanism and a chain selection rule that give eventual consensus and then add a finality gadget, a protocol that finalises blocks that the participants already agree on, to get provable finality. We present a finality gadget that works in a partially synchronous network model, GRANDPA (GHOST-based Recursive Ancestor De-

riving Prefix Agreement), where participants vote on their own view of which chain is best, and it tries to finalize on the prefix that 2/3 of participants agree on. This allows GRANDPA to finalize as many blocks as the participants agree on in one round, whether this is one block or thousands of blocks. GRANDPA works in the presence of up to 1/3 Byzantine actors.

A design goal of GRANDPA was to separate the finality gadget from block production as much as possible. As a result GRANDPA is somewhat flexible (see Section 5) and can be adapted to deal with scenarios such as:

- changing the underlying block production mechanism or chain selection rule,
- dropping below 2/3 voters online,
- GRANDPA being switched on or off, or
- delaying finality of a particular block until additional criteria are met.

GRANDPA is somewhat more complicated to implement than comparable protocols such as Tendermint, Casper FFG or HotStuff [11], while offering similar efficiency and so its flexibility is its key advantage.

Expressing the properties that GRANDPA satisfies formally is beyond the scope of this paper so we will discuss them informally here. With any underlying block production mechanism and even with no network guarantees, it satisfies *safety*, that all blocks finalised at any time by any participant lie on the same chain, and *validity*, that any block that is finalised has to have been seen to be in the best chain, according to the chain selection rule, that includes its parent by some honest participant at some time. The *liveness* property under a partially synchronous network is a bit more involved: if the block production mechanism satisfies that if GRANDPA does not finalise any new blocks for long enough, then all honest voters will eventually agree on a longer common prefix than the last finalised block, then GRANDPA will keep finalising new blocks forever. Thus, for instance, if the block production mechanism is proof-of-work, and the best chain is defined to be the longest chain that includes the last finalised block, then using many possible analyses of Bitcoin, it is easy to show that GRANDPA will keep finalising new blocks. This is a much stronger property than the "plausible liveness" shown for Casper FFG [4].

Note that a more comprehensive version of this paper is available online [8].

2 GRANDPA: Model and Protocol

In this section we describe the GRANDPA protocol. Let us assume we have a number of participants that we refer to as voters v, who want to agree on finality. We assume that more than 2/3 of voters are voting truthfully. The votes are gossiped to all other voters and so any vote that is sent to an honest voter will reach all voters. We assume the network delay is at most T, that means a vote will not take more that T to traverse the network an reach any other voter.

We write $B \geq B'$ if B is B' or a descendent of B'. For a set of votes V, we write $g(V)$ for the latest block B such that more than 2/3 of voters v have votes in V for blocks $\geq B$. This process is demonstrated in Figure 1, where arrows point to the blocks parent. We sum up the votes on a block and it's

descendants. Then if less than 1/3 of voters have more than one vote in V (which honest voters will not), then the set of blocks that have over 2/3 total votes will form a chain and $g(V)$ is the head of this chain. The basic idea behind

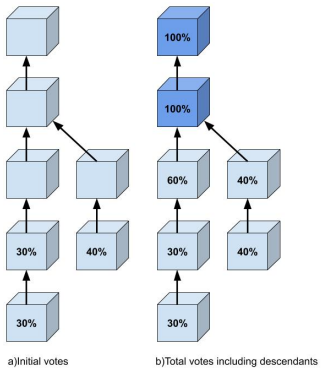

Figure 1: GRANDPA votes and how they are aggregated.

GRANDPA is that we run a protocol similar to other two-phase synchronous Byzantine agreement algorithms, except in how we use the function g on the set of votes in place of requiring agreement on an exact value. Effectively voters are voting on chains and not blocks. So where they would vote in the second phase on the exact value that over 2/3 of voters voted for in the first phase, GRANDPA used g on the first set of votes to find the longest common prefix that over 2/3 of voters agree on and use that for the second vote. Similarly after the second voting phase, instead of deciding the value that over 2/3 of voters voted on, GRANDPA again uses g to finalize the longest common prefix that over 2/3 voted for the the second phase. This means that even though participants might not agree on the latest block, they still may be able to agree on a long chain of earlier blocks.

We describe what participant v does in round r of GRANDPA. There are two voting phases in each round, *prevote* and *precommits*, and v maintains sets of prevotes $V_{r',v}$ and precommits $C_{r',v}$ it has seen from each round $r' \leq r$. It maintains an *estimate* $E_{r-1,v}(V_{r-1,v}, C_{r-1,v})$ of the last block that could have been finalised in round $r-1$, which we will define below. Note that we update this estimate upon receiving new votes, even from the previous round. It also has a notion of when a round is *completable*, which means that only blocks from one chain could be finalised in that round and so the above estimate is well-defined. Next we summarize GRANDPA in Algorithm 1 below.

Algorithm 1: GRANDPA Protocol

1. A voter v can start round $r > 1$ when round $r-1$ is completable and v has cast votes in all previous rounds where they are a voter. Let $t_{r,v}$ be the time v starts round r.

2. At time $t_{r,v}$, if v is the primary of this round then they broadcast $E_{r-1,v}(V_{r-1,v}, C_{r-1,v})$.

3. v waits until either it is at least time $t_{r,v} + 2T$ or round r is completable, then broadcasts a prevote. They prevote for the head of the best chain containing $E_{r-1,v}(V_{r-1,v}, C_{r-1,v})$ unless we received a block B from the primary and $g(V_{r-1,v}) \geq B > E_{r-1,v}(V_{r-1,v}, C_{r-1,v})$, in which case they use the best chain containing B instead.

4. v waits until $g(V_{r,v}) \geq E_{r-1,v}$ and either it is at least time $t_{r,v} + 4T$ or round r is completable, and then broadcasts a precommit for $g(V_{r,v})$.

v sees a block B as finalised in round r if $g(C_{r,v}) \geq B$.

3 GRANDPA's safety

Next we give the definitions of $E_{r-1,v}$ and completability and sketch how they ensure safety if more than 2/3 of voters are honest. Intuitively, a round r is completable when we are sure that only blocks from one chain could be finalised, and then $E_{r,v}$ is the last block on that chain that can be finalised. Because all votes are $\geq E_{r-1,v}$, then any block that could have been finalised in the last round is also finalised in this round. All blocks that are ever finalised in this round by any voter are on a single chain consisting of blocks $\leq g(C_r)$, where C_r is the precommits from round r seen by any voter at any time, this ensures safety i.e. that all blocks ever finalised by any honest voter are on a single chain.

In detail, $E_{r-1,v}$ is the earliest block $B \leq g(V_{r-1,v})$ that has that any possible child B' of B has that either $V_{r-1,v}$ or $C_{r-1,v}$ have that more than 2/3 if voters have votes that are not fr B' or its descendants. The round is completable when any block satisfies these conditions. For B' to be finalised by some voter v' in round $r-1$, then we would have $B' \leq g(C_{r-1,v'}) \leq g(V_{r-1,v'})$ and if $B' \leq E_{r-1,v}$, then it is not harder to show that at least 1/3 of voters either voted twice or precommited something other than $g(V_{r-1,V})$.

4 GRANDPA's liveness

By necessity, we can only sketch how the proof of liveness, which holds in a partially synchronous setting under the right assumptions about block generation and the fork choice used to select the prevotes. If at time t, an honest v sees sets of votes $V_{r-1,v}$ and $C_{r-1,v}$, then gossip ensures that at time $t+T$ all honest voters see supersets of these sets. This implies that if one voter sees round r as completable, then all do within time T. Since we skip all waiting steps when the current round is completable, this guarantees that round start tines are

within T. In turn, we can show that if the primary v is honest and broadcasts $B = E_{r-1,v}$, then all honest v' see that $E_{r-1,v'} \leq B \leq g(V_{r-1,v})$ and so everyone prevotes for the best chain given B. Whatever prefix the honest voters agree on will be finalised. If B is later than the last finalised block, then the last finalised block will advance. If B is the last finalised block for many (not-necessarily consecutive) rounds, then the underlying block production's eventual consensus will make honest voters agree on a longer prefix eventually.

5 Practical Advantages of GRANDPA

In this section we review the advantages of GRANDPA in terms of adaptability and flexibility, and efficiency.

a) *Making round times independent of block times*: In systems with Byzantine Agreement on every block such as Tendermint [3], block production has to be slower than reaching consensus. This is because every block needs at least two rounds of communication for being finalised before the next block can be produced. This means in the case we have many participants when the round time (which determines the time it takes to reach consensus) needs to be long, block production will be slow. However, there are systems that require reaching consensus is many times slower than block production such as Casper FFG. Here this is because rounds, which are referred to as a *checkpoints* are many blocks apart and hence a round takes many block times. This is done to optimise Casper FFG for a large number of participants. GRANDPA allows us to set round time independently of the block production time. By simply setting parameters we can deal with i) when block production is slower than reaching consensus or ii) when reaching consensus is slower than block production.

b) *Changing Consensus*: It makes easy to switch consensus. we can start with an eventual consensus protocol and we can add GRANDPA on top of it. Then GRANDPA will would finalise everything from the beginning. this would make it possible to make it possible to change the underlying eventual consensus protocol. For example, if there is proof-of-work, PoW, chain such as Bitcoin [7][6] that wants to change to a proof-of-stake, PoS, rule after while. It would be very difficult because we do not have a final block that would be the transition block, because a block is only even probably final when it has been built on. However, with GRANDPA this is possible.

c) *Allows Delaying Finality (tune-ably slow)*: there might be situations where it would be useful if we can delay finality. For example, in multi-chain systems all data is not on a single chain. A block producing mechanism could act on the data that we do not know is valid yet and GRANDPA would finalise them later when we know that the data we acted on is valid, hence GRANDPA allows for optimistic execution.

6 GRANDPA in Polkadot

GRANDPA is the finality gadget used for the Polkadot system [9]. GRANDPA is running on Polkadot testnets since October 2018 and is running currently on the recently launched "canary network" Kusama (https://kusama.network). It has been used with two block production protocols (Aura [1], BABE [2]) and

more are planned. It has been tested on testnets with up to 100 voters where it still was able to finalise blocks in under 2s.

Since the block times are longer than that, it would have been no less efficient to use a Byzantine Agreement on each block protocol, such as Tendermint. However, there are plans to expand the voter set to at least 1000 nodes, and make extensive use of GRANDPA's practical advantage (c) described in Section 5. GRANDPA is run by Polkadot's validator nodes to secure Polkadot's relay chain. Polkadot will have many other chains, called parachains, whose consensus is coordinated by the relay chain. Necessarily it requires more validators to scale to more parachains. The way Polkadot's data availability is tied to it's consensus requires all validators to be voters: erasure coded parachain data is distributed to all validators in such a way that 1/3 of validators can reconstruct it and so if these pieces are required for prevotes in GRANDPA, we can guarantee that this data is available if a block if finalised since the block received at least 1/3 votes from honest validators. In Polkadot, finality will be delayed to allow reports and checking of the validity and unavailability of parachain blocks. While these reports delay finality, messages between parachains will be able to be acted on in a single block. Thus Polkadot will be making extensive use of GRANDPA's optimistic execution property of separating block production from finality.

We would like to acknowledge Rob Habermeier from Parity Technologies for many discussions and useful suggestions and implementing this crazy protocol, Fatemeh Shirazi for her help with this manuscript, many other Parity Technology developers and Web3 Foundation researchers for their input.

References

[1] Aura. https://wiki.parity.io/Aura. Accessed: 2019-08-26.

[2] BABE. https://research.web3.foundation/en/latest/polkadot/BABE/Babe/. Accessed: 2019-08-26.

[3] Ethan Buchman, Jae Kwon, and Zarko Milosevic. The latest gossip on BFT consensus. *CoRR*, abs/1807.04938, 2018. http://arxiv.org/abs/1807.04938.

[4] Vitalik Buterin and Virgil Griffith. Casper the friendly finality gadget. *CoRR*, abs/1710.09437, 2017. http://arxiv.org/abs/1710.09437.

[5] Jing Chen and Silvio Micali. Algorand. *arXiv preprint arXiv:1607.01341*, 2016.

[6] 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.

[7] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. 2008.

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

[9] Gavin Wood. Polkadot: Vision for a heterogeneous multi-chain framework. *White Paper*, 2016. https://polkadot.network/PolkaDotPaper.pdf.

[10] Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger eip-150 revision (759dccd - 2017-08-07), 2017. Accessed: 2018-01-03.

[11] Maofan Yin, Dahlia Malkhi, Michael K Reiter, Guy Golan Gueta, and Ittai Abraham. Hotstuff: Bft consensus in the lens of blockchain. *arXiv preprint arXiv:1803.05069*. https://arxiv.org/abs/1803.05069.