Modern Byzantine Fault Tolerant Consensus and Applications

M1 Ma Yunzhe
School of Computing
Tokyo Institute of Technology

1

1

Overview

- No new consensus protocol presented today
- Introduce some improvements of BFT consensus
- How to select / design a BFT consensus

Name	Time	Features
PBFT	1999	First practical BFT consensus
SBFT	2018	Linear message complexity
Tendermint	2018	Gossip protocol and rotating leader
HotStuff	2019	Pipelined BFT consensus

2

2

Background

- Byzatine Fault Tolerance (BFT)
  - The Byzantine generals problem[1] – 1982
  - Distributed system with malicious components
- Blockchain
  - Permissionless blockchain
    - Proof-of-Work, Bitcoin
    - Proof-of-Stake
  - Permissioned blcokchain
    - BFT consensus

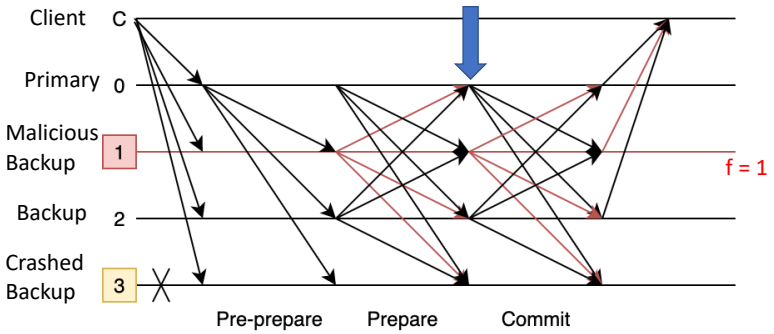
[1] LAMPORT, LESLIE, ROBERT SHOSTAK, and MARSHALL PEASE. "The Byzantine Generals Problem." *ACM Transactions on Programming Languages and Systems* 4.3 (1982): 382-401.

3

3

PBFT

- Practical Byzantine fault tolerance[2] – 1999
- Asynchronous network, such as the Internet
- N nodes, at most f malicious nodes, f ≤ (n-1)/3



[2] Castro, Miguel, and Barbara Liskov. "Practical Byzantine fault tolerance." *OSDI*. Vol. 99. No. 1999. 1999.

4

4

PBFT

- State machine replication
  - Network file system
  - Distributed database
- Permissioned blockchain

- Different requirements
  - Scalability / Consistency / Latency / Network
  - Need improvements to apply BFT consensus

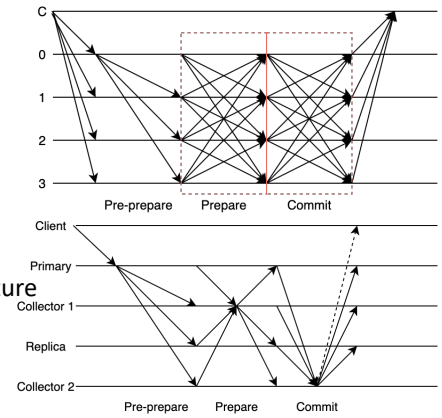
5

5

Message complexity

- Problem 1: Message complexity
  - PBFT - $O(n^2)$
  - Make PBFT linear

- $SBFT_{[3]}$ – 2018
- Introducing collector
  - Threshold signature
    - Reducing cost of signature
  - Redundant collectors
    - Avoiding SPOF



[3] Gueta, Guy Golan, et al. "SBFT: a Scalable and Decentralized Trust Infrastructure." *arXiv preprint arXiv:1804.01626* (2018).

6

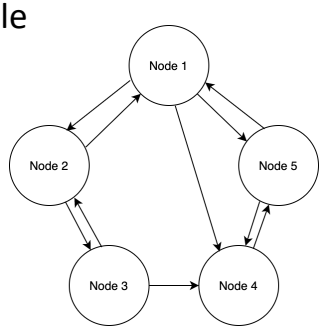
6

Network

- Problem 2: Network model
- Direct connection is not available

- Tendermint[4] – 2018
- Introducing gossip protocol

- Tradeoff: Communication time

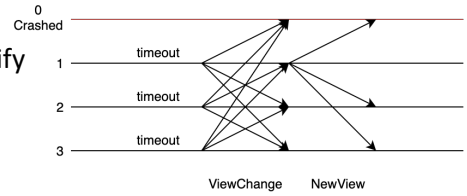


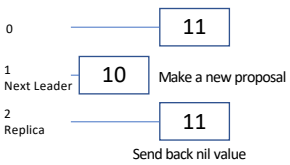
[4] Buchman, Ethan, Jae Kwon, and Zarko Milosevic. "The latest gossip on BFT consensus." *arXiv preprint arXiv:1807.04938* (2018).

7

Rotating leader

- Problem 3: Recovery mode
- Complicated Recovery mode
  - Special message
  - Exchange information to verify
  - Message complexity
- Tendermint[4] – 2018
- Introducing rotating leader
  - No recovery mode
    - Using locked round number to verify
    - Send nil value when verification failed
  - Change leader every round



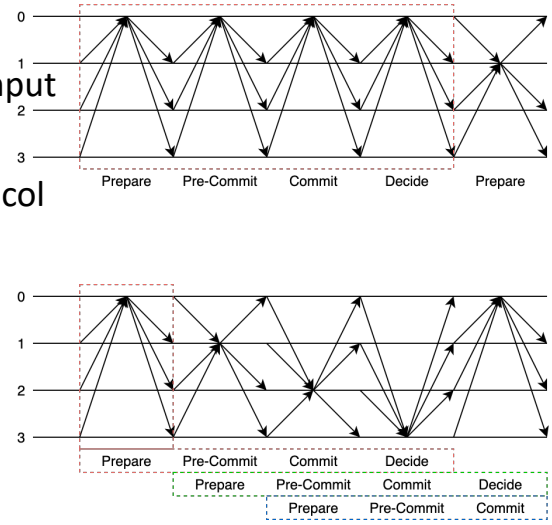


[4] Buchman, Ethan, Jae Kwon, and Zarko Milosevic. "The latest gossip on BFT consensus." *arXiv preprint arXiv:1807.04938* (2018).

8

Pipeline

- Problem 4: Throughput
- HotStuff[5] – 2019
- Pipelined BFT protocol
  - Collector
  - Rotating leader
  - No recovery mode
  - => Same pattern
- Tradeoff: Latency



[5] Yin, Maofan, et al. "Hotstuff: Bft consensus with linearity and responsiveness." *Prceedings of the 2019 ACM Symposium on Principles of Distributed Computing*. 2019.

9

9

Modern BFT consensus

- With blockchain
  - Blockchain consensus
    - Extend the use cases
  - Diverse requirements
    - Scalability / Consistency / Latency / Responsiveness / Network
  - Advanced cryptography
    - Threshold signature / Key aggregation / Signature aggregation

10

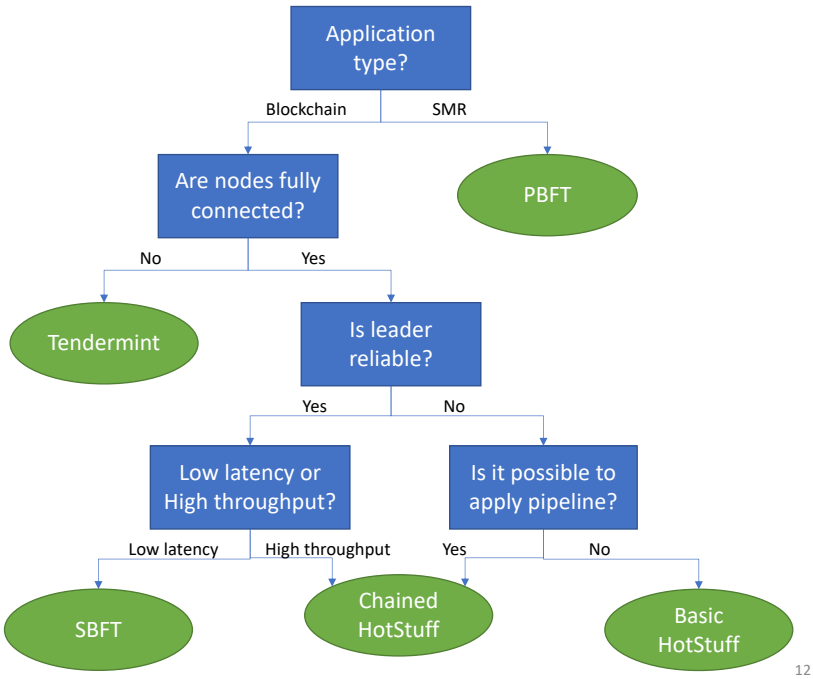
10

Conclusion

- It's hard to design a universal BFT consensus
- Based on the requirements
  - SMR / Blockchain / Just agreement / …
  - Network model
  - Message complexity
  - Stable leader / Rotating leader
  - Pipeline

11

11



12