Round-Efficient Byzantine Broadcast under Strongly Adaptive and Majority Corruptions

Jun Wan (junwan@mit.edu)
Hanshen Xiao (hsxiao@mit.edu)
Srini Devadas (devadas@csail.mit.edu)
Elaine Shi (runting@gmail.com)
Byzantine Broadcast [Lamport et al. 82]

- A set of users aim to reach consensus, one of them is the designated sender.
- The sender is given an input bit $b \in \{0, 1\}$
  - Consistency: all honest users must output the same bit; and
  - Validity: all honest users output the sender’s input bit if the sender is honest.
Background and Previous Work

- Under synchronous setting,
- [Dolev and Strong, 83]: no deterministic protocol can achieve Byzantine Broadcast within $f + 1$ rounds, where $f$ is the number of corrupted users.
- Focus on randomized protocols
Previous work

- Honest majority: expected constant rounds [Katz and Koo 09, Abraham et al. 19], even under a strongly adaptive adversary.
- Dishonest majority: expected constant rounds [Chan et al. 20, Wan et al. 20], but only under a weakly adaptive adversary.
Byzantine Broadcast: adversary model

Static: decide who to corrupt before the protocol.

Send: Hi, Dad! Everything is good!

Receive: Hi, Dad! Everything is good!
Byzantine Broadcast: adversary model

Weakly adaptive: can corrupt during the protocol, but any message sent in the round of corruption must be delivered.

Send: Hi, Dad! Every thing is good!
Send: I am sick! Need 1000 bucks!
Being sent: Hi, Dad! Every thing is good!
Receive: Hi, Dad! Every thing is good! I am sick! Need 1000 bucks!
Strongly rushing adaptive: can perform “after-the-fact removal”, can erase the messages any node had sent in the same round it became corrupt.
Is it possible to achieve sublinear round complexity under
- dishonest majority and
- a strongly rushing adaptive adversary?
Assuming the existence of a trusted setup and time-lock puzzles,

**Theorem**

There exists a protocol that achieves BB in \( \left( \frac{n}{n-f} \right)^2 \cdot \text{polylog}(\lambda) \) number of rounds with probability \( 1 - \text{negl}(\lambda) \).
Chan et. al.’s construction

- Define two committees: the 0-committee and the 1-committee.
- The $b$-committee consists of all nodes whose VRF evaluation on $b$ is smaller a parameter.
- Define two committees: the 0-committee and the 1-committee.
- The $b$-committee consists of all nodes whose VRF evaluation on $b$ is smaller a parameter.
- Parameters are chosen such that each committee’s size is polylogarithmic in expectation.
Chan et. al.’s construction

- Each committee’s size is polylogarithmic in expectation.
- Committee members engage in poly-logarithmically many rounds of voting.
- All nodes, including non-committee members, keep relaying the votes they have seen.
How to change it

- If only adversary cannot corrupt users before votes are delivered!
- How about we “encrypt” the votes:
  - Adversary need one round of time to “decrypt” the vote.
  - Non-committee members send dummy message.
A time-lock puzzle with parameter $\xi$ and $T$,

- For any message, can generate a puzzle in $\text{polylog}(T)$ steps.
- Given a puzzle, can solve it in $T$ steps under a sequential Random-Access Machine.
- Even parallel adversary cannot solve it in less than $\xi T$ steps.
Intuitions for using time-lock puzzle

- Voters lock the votes in a time-lock puzzle.
- Non-voters send chaff of the same length, also locked in puzzles.
- Even if the adversary has unbounded parallelism,
  - cannot distinguish voters and non-voters within one round of time.
Challenge

- Adversary has access to unbounded parallelism, but honest users don’t.
- Honest users do not have a consistent view of the puzzles being distributed.
  - Hard to coordinate who solves which puzzles.
The Age-based Sampling Protocol

Allow all honest users to each distribute a time-lock puzzle embedding some messages.

- **Liveness**: every honest node will receive the solution of all honest puzzles.
- **Momentary secrecy**: the adversary cannot learn any information about honest users’ encoded messages within one round of time,
  - even if it has unbounded parallelism.
Every user computes and sends a time-lock puzzle on the message.

Repeat $\Theta(\log n)$ iterations: in the $i^{th}$ iteration,
- each iteration has time $T_{\text{solve}} \cdot \text{polylog}(\lambda)$.
- for each unsolved puzzle, solve it with probability $\min(2^\alpha \cdot p, 1)$ where $p = \ln(16n/h)/n = \Theta(1/n)$.
  - $\alpha$ is the age of the puzzle.
- send solutions of newly solved puzzles to other users.
Proof of Correctness

- Liveness: at least half of the unsolved puzzles is solved by honest users in each iteration (no matter who the adversary corrupt).
- Momentary secrecy: follows from properties of time-lock puzzle.
- Failure probability for the $i^{th}$ iteration:

$$\left(1 - 2^i \cdot p\right)^{h \cdot n/2^{i-1}} \cdot \binom{n}{h} \cdot \binom{n}{n/2^{i-1}} \leq \exp(-\Theta(n)).$$
Round complexity: $E \cdot P \cdot \left( \frac{T_{solve}}{T_\emptyset} \right)$:

- $E$: number of epochs / iterations, $\Theta(\log n)$.
- $P$: number of puzzles an honest user need solve per iteration, upper bounded by $\text{polylog}(n, \lambda)$ by Chernoff Bound.
- $T_{solve}$: time to solve each puzzle.
- $T_\emptyset$: time per round.

By definition of time-lock puzzle: $\left( \frac{T_{solve}}{T_\emptyset} \right)$ is upper bounded by $1/\xi$. 
Apply the Age-based Sampling Protocol

- Combine it with techniques proposed in Chan et al. [CPS20]:
  - replace normal message relay with Distribute protocol.
- Most challenging aspect: how to prove security.
We propose a Byzantine Broadcast protocol under
- dishonest majority
- a strongly adaptive adversary.
- round complexity: polylogarithmic.
Open questions

Is it possible to achieve Byzantine Broadcast under a strongly adaptive adversary in expected constant rounds?
Acknowledgement

Thank you.