Computer Science and Artificial Intelligence Laboratory

Technical Report

MIT-CSAIL-TR-2017-004 March 31, 2017

Optimal and Player-Replaceable Consensus with an Honest Majority

Silvio Micali and Vinod Vaikuntanathan

massachusetts institute of technology, cambridge, ma 02139 usa — www.csail.mit.edu

Optimal and Player-Replaceable Consensus with an Honest Majority

Silvio Micali* MIT

Vinod Vaikuntanathan† MIT

March 22, 2017

Abstract

We construct a Byzantine Agreement protocol that tolerates $t < n/2$ corruptions, is very efficient in terms of the number of rounds and the number of bits of communication, and satisfies a strong notion of robustness called player replaceability (defined in [Mic16]). We provide an analysis of our protocol when executed on real-world networks such as the ones employed in the bitcoin protocol.

*E-mail: silvio@csail.mit.edu

†E-mail: vinodv@csail.mit.edu

1 Introduction

In 1980, Pease, Shostak and Lamport [PSL80] defined the problem of Byzantine agreement, and since then, it has arguably become the central problem in distributed computation tolerating faulty behavior. Informally, the problem is to maintain a common view of the world in the presence of faulty processes that strive to prevent the good processes from reaching agreement. The faults are "Byzantine" in the sense that their strategy is arbitrary and is focused on maximizing the chance of disagreement among the honest players (and, except when we explicitly say so, is also polynomial-time bounded).

Definition 1.1. A protocol among n players, in which each player starts with an input is a Byzantine Agreement protocol with t faulty players, if the following conditions hold:

- Completeness (Validity): If the input values of all the honest players are the same (say, a value v), then the output of all the honest players is v itself.
- Soundness (Agreement): No two honest players have different outputs, regardless of the inputs of the players and regardless of the strategy employed by the t dishonest players.

We work in the standard synchronous communication model. Here, Pease, Shostak and Lamport [PSL80] showed a protocol with round complexity of $t + 1$ rounds for any $t < n/3$ faulty players, which was shown to be optimal for deterministic protocols by Fischer and Lynch [FL82]. However, the communication complexity of the protocol was exponential in n. Following a series of works, Garay and Moses [GM98] constructed a deterministic BA protocol that runs for $t + 1$ rounds, with a polynomial communication.

Faced with the lower bound on the round complexity for deterministic protocols, the natural direction of research was to find ways to overcome this limitation, the first choice being to resort to randomization. This direction was pursued early on, starting with the work of Ben-Or and Rabin [Ben83, Rab83] who showed how to reach Byzantine agreement quickly given a source of "common coins". Thus, the bulk of the attention was concentrated on constructing protocols that generate a common-coin in a network. This line of research culminated in the groundbreaking work of Feldman and Micali [FM97], who designed a protocol to generate a common coin in *expected* $O(1)$ rounds and with polynomial communication, under the assumption that the point-to-point channels connecting pairs of processors are private. This, in turn, gave Byzantine Agreement protocols that run in expected $O(1)$ rounds with $t < n/3$ faulty players. In all this work, the processors could have arbitrary, potentially computationally unbounded, strategies.

Building on this, Katz and Koo [KK06] showed a protocol with expected $O(1)$ rounds tolerating $t < n/2$ faulty players, assuming that the processors are computationally bounded and that secure digital signature schemes exist. Finally, in a recent work as part of the Algorand shared ledger protocol, Micali [Mic16] showed an entirely new and surprisingly simple Byzantine agreement protocol with $t < n/3$ faulty players.

Our Results. We show a new Byzantine agreement protocol tolerating $t < n/2$ faulty players, improving on [KK06, Mic16] in several ways. Our protocol has a higher fault-tolerance, namely $t < n/2$, compared to [Mic16].

Our protocol improves on [KK06] in two significant ways. First, the protocol is more efficient. The round-complexity of the Katz-Koo protocol is a large constant owing to its use of sub-optimal primitives (starting from basic primitives such as graded broadcast [FM97]) and its reliance on expensive verifiable secret-sharing machinery. Secondly, their concrete communication complexity

when performing BA for large messages is sub-optimal as well. Our protocol improves [KK06] on both fronts. Concretely, for every $k \in \mathbb{N}$, our protocol is guaranteed to halt with agreement in $2k+3$ rounds with probability $1 - 2^{-k}$. Of these rounds, only three of them involve sending large messages while the rest only involve communicating short signatures of around 300 bits. Furthermore, we provide an analysis of the cost of the protocol in a real-world propagation network (such as the one used in the design of shared public ledgers [Nak, Mic16]).

Secondly, and more importantly, the protocol satisfies the strong security notion of player replaceability, first defined in [Mic16]. Roughly speaking, consider the protocol executing over a very large network of n players where in each round r, some (uniformly random) small subset of players is chosen according to some external protocol to carry out the r^{th} round of the protocol. Once these players are done, they pass the baton to the next set and they could all be immediately corrupted by the adversary. In other words, this entails being robust to a very strong notion of adaptive corruption. The players running each round of the protocols are *replaceable*, in fact necessarily so as their identities are unknown up until the r^{th} round and they are corrupted immediately after.

The protocol of [KK06] does not satisfy this strong notion of security. The protocol of [Mic16] does, but only for $t < n/3$ corruptions. Our protocol achieves the best of both worlds, achieving player replaceable BA with $t < n/2$ corruptions.

Organization of the Paper. We describe our graded broadcast protocols in Sections 3 and 4. Finally, we put these together to construct our eventual BA protocol in Section 5.

2 Definitions

2.1 Byzantine Agreement and Friends

Byzantine Agreement. The problem of Byzantine Agreement [PSL80] is as defined below.

Definition 2.1 (Byzantine Agreement)**.** Let $\mathcal{D}$ be a finite set. Let Π be a protocol among n players, in which each player P_i starts with an input value $v_i \in \mathcal{D}$, and outputs a value w_i at the end of the protocol. Π is a Byzantine Agreement protocol, if the following conditions hold:

1. (AGREEMENT) For any two non-faulty players P_i and P_j, $w_i = w_j$.
2. (VALIDITY) If all the non-faulty players have the same input v, then the output w_i of every non-faulty player P_i is v itself.
3. (TERMINATION) Protocol Π terminates eventually. The probability that Π has not terminated within t rounds is a vanishing function of t. More precisely,

$$\lim_{t \to \infty} \Pr[\Pi \text{ has not terminated in } t \text{ rounds}] = 0$$

   where the probability is over the coin-tosses of all the non-faulty players.

When $|\mathcal{D}| = 2$, then the problem is called binary Byzantine Agreement.

If Π is a randomized protocol, then the Agreement and the Validity conditions are required to hold *with probability* 1 over the coin-tosses of the processors. The principal complexity measure of interest is the *expected* running time of the protocol.

Graded Broadcast. Since we do not have built-in broadcast channels in the real world, we would like to *simulate* a broadcast channel using a protocol among the players. As a useful intermediate step to achieving fully reliable broadcast, we would like to define a weaker notion of broadcast that is nevertheless very easy to achieve. Informally, the right notion is that of a "semi-reliable" broadcast channel that loses messages sometimes, but never delivers two different messages to two different players.

There are various definitions of such a semi-reliable broadcast channel in the literature, starting with the work of [] who defined what they called a Crusader agreement, and the work of Feldman and Micali [] who defined the notion of graded broadcast. What we use here is the Feldman-Micali notion of graded broadcast, as well as stronger variants thereof.

Informally, a graded broadcast protocol is a protocol with a designated player called "dealer" (the one who broadcasts) such that:

- If the dealer is good, all the players get the same message.
- Even if the dealer is bad, if some good player accepts the message, all the good players get the same message (but they may or may not accept it).

Formally, the notion of gradecast is as follows.

Definition 2.2 (Graded Broadcast [FM97])**.** A protocol Π is said to be achieve graded broadcast if, at the beginning of the protocol, a designated player D (called *the dealer*) holds a value v, and at the end of the protocol, every player P_i outputs a pair (v_i, c_i) such that the following properties hold: ($\forall i$, $c_i \in \{0, 1, 2\}$)

1. If D is honest, then $v_i = v$ and $c_i = 2$ for every honest player P_i.
2. For any two honest players P_i and P_j, $|c_i - c_j| \leq 1$.
3. (Consistency) For any two honest players P_i and P_j, if $c_i > 0$ and $c_j > 0$, then $v_i = v_j$.

Feldman and Micali [FM97] also constructed a constant-round deterministic protocol which solves the graded broadcast problem – they called their protocol a "gradecast" protocol. An $O(1)$-rounds deterministic protocol with these guarantees appears, for instance, in Feldman and Micali [FM97] as a "gradecast" protocol. Katz-Koo [KK06] constructed a graded broadcast protocol with $t < n/2$ corruptions with XX rounds.

2.2 Probabilistic Tools

We use the following version of Chernoff Bound.

Proposition 2.3 (Chernoff Bound)**.** Let $X_1, X_2, \ldots, X_n$ be independent Poisson trials such that, for $1 \leq i \leq n$, $\Pr[X_i = 1] = p_i$, where $0 < p_i < 1$. Then, for $X = \sum_{i=1}^{n} X_i$, $\mu = \mathbb{E}[X] = \sum_{i=1}^{n} p_i$, and any $\delta > 0$, $\Pr[X > (1+\delta)\mu] < \left[\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right]^{\mu}$. In particular, if $\delta = 2$, we get $\Pr[X > 3\mu] < e^{-\mu}$.

3 Protocol for $\{0, 1\}$-Graded Broadcast when $t < n/2$

Let the set of players be $\mathcal{P}$. The protocol Π_{GC01} proceeds as follows. The Sender S starts the protocol with input x and all other players start empty-handed.

1. Sender S: Broadcast input x together with signature $(x, \mathsf{Sig}_S(x))$.

2. Player P: If P sees x with valid signature from S, P broadcasts $(x, \mathsf{Sig}_S(x))$. Else, send nothing.

3. Player P (Local): If P sees $> n/2$ players with the same x and valid signatures and no player with a different x' and valid signature, then P sets $x_P = x$ and $\mathsf{grade}_P = 1$. Else, P sets $x_P = \perp$ and $\mathsf{grade}_P = 0$.

Theorem 3.1. Π_{GC01} *is a* 2 *round* $\{0,1\}$*-gradecast protocol.*

Proof. We show completeness (validity), soundness (agreement) and discuss both the number of rounds and the (wall-clock) time.

Completeness (Validity). If the sender S is honest, all honest players broadcast $(x, \mathsf{Sig}_S(x))$ in Step 2. In (Local) Step 3, all honest players see $> n/2$ copies of x with valid signatures and, by security of the digital signature, no $x' \neq x$ with a valid signature.

Consequently, each honest player P sets $x_P = x$ and $\mathsf{grade}_P = 1$.

Soundness (Agreement). Suppose an honest player P sets $x_P = x$ and $\mathsf{grade}_P = 1$. This is because P sees (in Step 3) $> n/2$ valid signatures of x and no valid signature of any $x' \neq x$. One of the valid signatures of x comes from an honest player who necessarily sends it to all players in Step 2.

Each other honest player Q thus sees a valid signature of x, together with potentially valid signatures of other $x' \neq x$. Thus, his only options are (a) to output $x_Q = x$ and $\mathsf{grade}_Q = 1$, or (b) to output $x_Q = \perp$ and $\mathsf{grade}_Q = 0$.

In summary, if $\mathsf{grade}_Q = 1$, then $x_Q = x_P$, which is the definition of soundness for $\{0,1\}$-graded broadcast.

Rounds and Time. Clearly, the protocol takes two rounds. The total time is the time to twice propagate the value x together with a signature (32 bytes), in other words, two short rounds. The total time is $2 \times 1\mathsf{sec} = 2\mathsf{sec}$.

□

4 Protocol for $\{0,1,2\}$-Graded Broadcast when $t < n/2$

The protocol Π_{GC012} proceeds as follows. The Sender S starts the protocol with input x and all other players start empty-handed.

1. Sender S: Broadcast input x with signature $(x, \mathsf{Sig}_S(x))$.

2. Player P: If P sees x with valid signature from S, P countersigns and broadcasts $(x, \mathsf{Sig}_P(\mathsf{Sig}_S(x))$. Else, send nothing.

3. Player P: If P sees $> n/2$ valid countersignatures $(x, \mathsf{Sig}_i(\mathsf{Sig}_S(x))$ for some x, and *no* contradiction (i.e., no $(x', \mathsf{Sig}_i(\mathsf{Sig}_S(x'))$ for any $x' \neq x$), then broadcast

$$\left(x, \mathsf{SIGSET}_P(x)) := \left\{i, \mathsf{Sig}_i(\mathsf{Sig}_S(x))\right\}_i\right)$$

Otherwise broadcast nothing.

A SIGSET is defined to be consistent if it has $> n/2$ valid countersignatures.

4. Player P (Local):

   - IF P sees $> n/2$ consistent SIGSETs for the same x, and no consistent SIGSET for any $x' \neq x$ THEN output $x_P = x$ and $\mathsf{grade}_P = 2$.
   - IF P sees any consistent SIGSET for x, and no consistent SIGSET for $x \neq x$, THEN output $x_P = x$ and $\mathsf{grade}_P = 1$.
   - ELSE output $x_P = \perp$ and $\mathsf{grade}_P = 0$.

Theorem 4.1. Π_{GC012} *is a* 3 *round* $\{0,1,2\}$*-gradecast protocol.*

Proof. We show completeness (validity), soundness (agreement) and discuss both the number of rounds and the (wall-clock) time.

Completeness (Validity). If the sender S is honest, then all honest players P broadcast $(x, \mathsf{Sig}_P(\mathsf{Sig}_S(x)))$ in Step 2. In Step 3, all honest players see $> n/2$ valid countersignatures of x and, by the security of the digital signature, no valid countersignature of any $x' \neq x$. Thus, all honest players P broadcast $(x, \mathsf{SIGSET}_P(x))$. In (Local) Step 4, all honest players see $> n/2$ consistent SIGSETs of x. Consequently, each honest player P sets $x_P = x$ and $\mathsf{grade}_P = 2$.

Soundness (Agreement). Assume that an honest player P outputs $x_P = x$ and $\mathsf{grade}_P = 2$. We wish to argue that for every honest player Q, $x_Q = x$ and $\mathsf{grade}_Q \geq 1$.

Q sees x with at least one consistent SIGSET. It suffices to argue that it does not see an $x' \neq x$ with a consistent SIGSET. Suppose, for contradiction, that it did. This SIGSET contains $> n/2$ countersignatures of x', at least one of which belongs to an honest party. This honest party would then have sent this countersignature to everyone in Step 2. Since all honest parties see a countersignature of x', none of them would transmit a consistent SIGSET for x. Thus, there is no way that P saw $> n/2$ consistent SIGSETs for x, contradicting the assumption that P outputs $x_P = x$ with $\mathsf{grade}_P = 2$.

Rounds and Time. Clearly, the protocol takes 3 rounds. Round 3 is a "large round" that costs 50sec. The total time is $2 \times 1 + 50\mathsf{sec} = 52\mathsf{sec}$.

□

5 Protocol for Byzantine Agreement (for many bits) with $t < n/2$

We first present a tightly scheduled version of Byzantine Agreement.

- (Super-Step 1) Sender S runs a $\{0,1,2\}$-graded broadcast of his input x. Each player P gets x_P together with a value $\mathsf{grade}_P \in \{0,1,2\}$. Each player P sets

$$b_P = \begin{cases} 0 & \text{if } \mathsf{grade}_P = 2 \\ 1 & \text{if } \mathsf{grade}_P < 2 \end{cases}$$

  For K rounds, repeat the next three steps:

- (Super-Step 2) Each player P runs a $\{0,1\}$-graded broadcast with input b_P.
- (Coin) All players P interact, in one round, to generate a coin using the protocol of **[?]**.

- (Local Computation) If more than $n/2$ graded broadcasts (from Super-Step 2) resulted in a bit b with $\mathsf{grade}_P = 1$, set $b_P = b$. Else, set $b_P = \mathsf{coin}_P$.

  Go back to 2a.

- (Local Output Step) If $b_P = 0$, output x_P and if $b_P = 1$, output $\perp$.

Proof. We prove completeness, soundness and analyze the running time.

Completeness (Validity). If the sender S is honest, then all honest players P obtains $x_P = x_S$ and $\mathsf{grade}_P = 2$ at the end of super-step 1. Thus, they all run super-step 2, with input $b_P = 0$ and do not change b_P during the k iterations. Finally, once all k iterations are over, they will have $b_P = 0$ and thus will output x_P which is equal to x_S.

Soundness (Agreement). Consider any iteration i of the loop at the end of which all honest players set b_P to be the same bit. We first claim that for all honest players, b_P will remain the same in all subsequent iterations. This is because in iteration $i+1$, they all start with the same bit and by the property of $\{0,1\}$-gradecast, all honest players will see $> n/2$ graded broadcasts with $\mathsf{grade} = 1$, meaning they will not change b_P from then on.

We now want to bound the probability of disagreement at the end of the protocol. Let E_i be the event that in iteration i, all honest players set b_P to be the same bit, and let E be the event that there is an iteration where all honest players set b_P to be the same bit. We now bound the probability that E does not happen.

$$\Pr[E_i] = \Pr[E_i|\text{ bad leader }]Pr[\text{bad leader}] + Pr[E_i|\text{ good leader }]Pr[\text{good leader}] \geq 1/2 \cdot \alpha$$

where α is the fraction of honest players.

Thus, the probability that none of the events E_i happen is at most $(1-\alpha/2)^k$. For an error probability of ϵ, then

$$k = \log(1/\epsilon)/\log(2/2-\alpha)$$

OPTIMISTIC ANALYSIS: if $\epsilon = 1/10^{12}$ and $\alpha = 0.8$ then $k <= 12/0.222 = 54$.

Rounds and Time. We first claim that super-step 2 and the coin step can be run concurrently for a total of 2 rounds per iteration. These rounds require honest players to exchange single signatures (so they are "short" rounds). The total is $3+2k$ rounds. The total time is $52+2k$ sec.
OPTIMISTIC ANALYSIS: for the numbers as above, the total time is $52+108 = 160$sec.

□

References

[Ben83] Michael Ben-Or. Another advantage of free choice: Completely asynchronous agreement protocols (extended abstract). In *PODC*, pages 27–30, 1983.

[FL82] Michael J. Fischer and Nancy A. Lynch. A lower bound for the time to assure interactive consistency. *Inf. Process. Lett.*, 14(4):183–186, 1982.

[FM97] Pesech Feldman and Silvio Micali. An optimal probabilistic protocol for synchronous byzantine agreement. *SIAM J. Comput.*, 26(4):873–933, 1997.

[GM98] Juan A. Garay and Yoram Moses. Fully polynomial byzantine agreement for n $>$ 3t processors in t + 1 rounds. *SIAM J. Comput.*, 27(1):247–290, 1998.

[KK06] Jonathan Katz and Chiu-Yuen Koo. On expected constant-round protocols for byzantine agreement. *ECCC Report TR06-028*, 2006.

[Mic16] Silvio Micali. ALGORAND: the efficient and democratic ledger. *CoRR*, abs/1607.01341, 2016.

[Nak] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, http://bitcoin.org/bitcoin.pdf.

[PSL80] M. Pease, R. Shostak, and L. Lamport. Reaching agreement in the presence of faults. *Journal of the ACM.*, 27:228–234, 1980.

[Rab83] Michael O. Rabin. Randomized byzantine generals. *FOCS*, pages 403–409, 1983.