

# Byzantine Agreement with Incomplete Views

Hanjun Li

December 2, 2019

# The Byzantine Agreement (BA) Problem

## Definition (Binary Byzantine Agreement)

Suppose there are  $N$  parties

- $P_1, \dots, P_N$ . Each  $P_i$  holds an initial input  $v_i \in \{0, 1\}$ ,
- $t$  of the parties are corrupted by an adversary.

A protocol achieves Byzantine Agreement if the following conditions hold after termination:

- (Validity) If all honest parties begin with the same input  $v$ , they also output  $v$ .
- (Agreement) All honest parties output the same value

# Standard Model

## Communication

- All parties communicate in rounds over point-to-point channels
- The communication network is *complete*

# Standard Model

## Communication

- All parties communicate in rounds over point-to-point channels
- The communication network is *complete*

## Adversary

- Adaptively corrupt honest parties at the start of any round
- Corrupted parties deviates from the protocol arbitrarily
- Receives honest messages before sending corrupted messages

# Our Model

## Communication

- All parties communicate in rounds over point-to-point channels
- The communication network is *sparse*

# Our Model

## Communication

- All parties communicate in rounds over point-to-point channels
- The communication network is *sparse*

## Parametrization

(The inclusive neighborhood  $\Gamma_i$  of  $P_i$  is called its *view*)

- $\alpha$ , the maximum fraction of corruption in an honest view
- $\delta$ , the minimum fraction of overlapping between any pair of honest views

# BA in the plain model

## Corruption

- BA is possible if and only if less than  $1/3$  parties are corrupted. [PSL80], [LSP].

## Round Complexity

- There exists a BA protocol of  $t + 1$  rounds [GM98].
- No deterministic protocol of less than  $t + 1$  rounds exists [FL82].

# BA with PKI and randomization

## w/ PKI

- tolerates  $1/2$  corruption
- requires  $t + 1$  rounds  
[DS83]



# BA with PKI and randomization

## w/ PKI

- tolerates  $1/2$  corruption
- requires  $t + 1$  rounds [DS83]

## w/ Randomization

- tolerates  $1/3$  corruption
- runs in expected *constant* round [FM97]

# BA with PKI and randomization

## w/ PKI

- tolerates  $1/2$  corruption
- requires  $t + 1$  rounds [DS83]

## w/ Randomization

- tolerates  $1/3$  corruption
- runs in expected *constant* round [FM97]

## w/ PKI and Randomization

- tolerates  $1/2$  corruption
- runs in expected *constant* round [FG03], [KK06], [MV17]

# Our Result

- assumes a PKI setup
- uses randomization
- recall  $\delta = \text{overlapping}$ ,  $\alpha = \text{corruption}$

# Our Result

- assumes a PKI setup
- uses randomization
- recall  $\delta = \text{overlapping}$ ,  $\alpha = \text{corruption}$

## Theorem (Positive)

*If  $\delta > 2\alpha$ , there exists an expected  $O(n)$  round Byzantine Agreement protocol. Further if  $\alpha = 1/2 - \epsilon$  for any constant  $\epsilon$ , there exists an expected constant round Byzantine Agreement protocol in our model.*

# Our Result

- assumes a PKI setup
- uses randomization
- recall  $\delta = \text{overlapping}$ ,  $\alpha = \text{corruption}$

## Theorem (Positive)

*If  $\delta > 2\alpha$ , there exists an expected  $O(n)$  round Byzantine Agreement protocol. Further if  $\alpha = 1/2 - \epsilon$  for any constant  $\epsilon$ , there exists an expected constant round Byzantine Agreement protocol in our model.*

## Theorem (Negative)

*If  $\alpha \geq 1/2$  or  $\delta \leq 2\alpha$ , there does not exist a Byzantine Agreement protocol in our model*

# Unique Digital Signature: Definition

## Definition

A Digital Signature scheme is a triple of polynomial time computable algorithms (Gen, Sign, Verify).

- $(S_k, P_k) \xleftarrow{R} \text{Gen}(1^k).$
- $\sigma = \text{Sign}(S_k, x).$
- $b \xleftarrow{R} \text{Verify}(P_k, x, \sigma).$

# Unique Digital Signature: Definition

## Definition

A Digital Signature scheme is a triple of polynomial time computable algorithms (Gen, Sign, Verify).

- $(S_k, P_k) \xleftarrow{R} \text{Gen}(1^k)$ .
- $\sigma = \text{Sign}(S_k, x)$ .
- $b \xleftarrow{R} \text{Verify}(P_k, x, \sigma)$ .

## Correctness

With overwhelming probability over  $(S_k, P_k) \xleftarrow{R} \text{Gen}(1^k)$ :  
 $1 \xleftarrow{R} \text{Verify}(P_k, x, \text{Sign}(S_k, x))$ .

# Unique Digital Signature: Uniqueness

## Uniqueness

- There do not exist values  $(P_k, x, \sigma_1, \sigma_2)$  such that  $\sigma_1 \neq \sigma_2$ , and  $\text{Verify}(P_k, x, \sigma_1) = \text{Verify}(P_k, x, \sigma_2) = 1$ .
- Construction given in [MRVil] under the RSA assumption.



# Unique Digital Signature: Security

## Security (existentially unforgeable)

Let  $T$  be any polynomial time algorithm. The probability that  $T$  wins the following game must be negligible:

- Run  $(P_k, S_k) \xleftarrow{R} \text{Gen}(1^k)$ :
- Run  $(x, \sigma) \xleftarrow{R} T^{\text{Sign}(S_k, \cdot)}(1^k, P_k)$
- $T$  wins if  $1 \xleftarrow{R} \text{Verify}(P_k, x, \sigma)$

# PKI for our model

## Public-key Infrastructure

- $P_i$  is assigned  $Pk_i, Sk_i$
- $P_i$  is assigned  $Pk_j$  for all  $P_j$  in its view
- The adversary knows  $Pk_i$  for all  $P_i$

# Random Oracle

- We assume a public random oracle function  $H(\cdot)$  for simplicity
- In our protocol,  $H(\cdot)$  can be replaced by a Verifiable Random Function [MRVil]

## Definition (Random Oracle)

A random oracle  $H$  is a map from  $\{0, 1\}^*$  to  $\{0, 1\}^k$  chosen by selecting every bit of  $H(x)$  uniformly and independently, for every  $x$ .

# Road Map

- **Graded Broadcast**: Prevents a party from sending different values in the same round.
- **Leader Selection**: Guarantees honest parties to agree on an honest leader with some chance.
- **Main protocol**: Guarantees honest parties to either agree on the same value or follow its leader.

# Road Map

- **Graded Broadcast**: Prevents a party from sending different values in the same round.
- **Leader Selection**: Guarantees honest parties to agree on an honest leader with some chance.
- **Main protocol**: Guarantees honest parties to either agree on the same value or follow its leader.

## The size of honest views

For simplicity, we assume that all honest views have the same size  $n$ . However, all our results hold without this assumption.

# Graded Broadcast: Definition

## Definition (Graded Broadcast)

For  $N$  parties  $P_1, \dots, P_N$ , one of which is the dealer  $P_d$  holding a message  $m$ , the following must hold after the protocol:

- Each honest  $P_i \in \Gamma_d$  outputs  $(m_i, g_i)$ , where  $g_i \in \{0, 1\}$ .

- asdf
- asdf

# Graded Broadcast: Definition

## Definition (Graded Broadcast)

For  $N$  parties  $P_1, \dots, P_N$ , one of which is the dealer  $P_d$  holding a message  $m$ , the following must hold after the protocol:

- Each honest  $P_i \in \Gamma_d$  outputs  $(m_i, g_i)$ , where  $g_i \in \{0, 1\}$ .
- (Validity) If  $P_d$  is honest, then  $m_i = m$ , and  $g_i = 1$  for all honest  $P_i \in \Gamma_d$ .
- (Agreement) If two honest parties  $P_i, P_j \in \Gamma_d$  outputs  $(m_i, 1)$ , and  $(m_j, 1)$ , then  $m_i = m_j$ .

- asdf

- asdf

# Graded Broadcast: Protocol

## Protocol: Graded Broadcast w/ dealer $P_d$

- 1:  $P_d$  sends  $(m, \sigma_d(m))$  to all  $P_i \in \Gamma_d$
- 2: Every honest  $P_i \in \Gamma_d$  receives  $(m, \sigma_d(m))$ 
  - if  $\sigma_d(m)$  is valid, then  $P_i$  forwards it to all  $P_j \in \Gamma_i$
  - else  $P_i$  does nothing.



# Graded Broadcast: Protocol

## Protocol: Graded Broadcast w/ dealer $P_d$

- 1:  $P_d$  sends  $(m, \sigma_d(m))$  to all  $P_i \in \Gamma_d$
- 2: Every honest  $P_i \in \Gamma_d$  receives  $(m, \sigma_d(m))$ 
  - if  $\sigma_d(m)$  is valid, then  $P_i$  forwards it to all  $P_j \in \Gamma_i$
  - else  $P_i$  does nothing.
- 3: Every honest  $P_i \in \Gamma_d$  receives  $(m_j, \sigma_d(m_j))$  from some  $P_j \in \Gamma_i$ 
  - if there are contradicting valid  $\sigma_d(m), \sigma_d(m')$  and  $m \neq m'$ , then  $P_i$  outputs  $(\phi, 0)$ .
  - if there are at least  $(\delta - \alpha)n$  valid signatures on some  $m$ , then  $P_i$  outputs  $(m, 1)$
  - else,  $P_i$  outputs  $(\phi, 0)$ .

# Graded Broadcast: Proof

## Claim (Validity)

*If the dealer  $P_d$  is honest with message  $m$ , then all honest  $P_i \in \Gamma_d$  output  $(m, 1)$ .*

# Graded Broadcast: Proof

## Claim (Validity)

*If the dealer  $P_d$  is honest with message  $m$ , then all honest  $P_i \in \Gamma_d$  output  $(m, 1)$ .*

- no contradicting signature can be forged

# Graded Broadcast: Proof

## Claim (Validity)

*If the dealer  $P_d$  is honest with message  $m$ , then all honest  $P_i \in \Gamma_d$  output  $(m, 1)$ .*

- no contradicting signature can be forged
- enough honest parties in  $\Gamma_d \cap \Gamma_i$  for all honest  $P_i \in \Gamma_d$

# Graded Broadcast: Proof

## Claim (Agreement)

*If two honest  $P_i, P_j \in \Gamma_d$  outputs  $(m_i, 1)$  and  $(m_j, 1)$ , and if  $\delta > 2\alpha$ , then  $m_i = m_j$ .*

# Graded Broadcast: Proof

## Claim (Agreement)

*If two honest  $P_i, P_j \in \Gamma_d$  outputs  $(m_i, 1)$  and  $(m_j, 1)$ , and if  $\delta > 2\alpha$ , then  $m_i = m_j$ .*

- $P_i$  receives forwarding  $(m_i, \sigma_d(m_i))$  from at least one honest party  $P_k \in \Gamma_d$

# Graded Broadcast: Proof

## Claim (Agreement)

*If two honest  $P_i, P_j \in \Gamma_d$  outputs  $(m_i, 1)$  and  $(m_j, 1)$ , and if  $\delta > 2\alpha$ , then  $m_i = m_j$ .*

- $P_i$  receives forwarding  $(m_i, \sigma_d(m_i))$  from at least one honest party  $P_k \in \Gamma_d$
- $P_k$  must have forwarded  $(m_i, \sigma_d(m_i))$  to  $P_j$ . Contradiction.

# Graded Broadcast: Summary

## Lemma (Graded Broadcast)

*If  $\delta > 2\alpha$ , there exists a three round Graded Broadcast protocol.*



# Leader Selection: Definition

## Definition (Leader Selection)

For a set of parties  $P_1, \dots, P_N$  and fairness  $\gamma$ , when the protocol terminates, the following conditions must hold with probability at least  $\gamma$ :

- Every honest  $P_i$  outputs  $P_I$  and  $P_I$  is honest by the end of the protocol.

When such an event happens, we say that an honest leader  $P_I$  is elected.

- asdf
- asdf

# Leader Selection: Protocol

## Protocol: Leader Selection w/ input $r$

- 1: Every honest  $P_i$  sends  $m_i = (i, \sigma_i(r))$  to all  $P_j \in \Gamma_i$
- 2: Every honest  $P_i$  forwards valid messages to all  $P_j \in \Gamma_j$

# Leader Selection: Protocol

## Protocol: Leader Selection w/ input $r$

- 1: Every honest  $P_i$  sends  $m_i = (i, \sigma_i(r))$  to all  $P_j \in \Gamma_i$
- 2: Every honest  $P_i$  forwards valid messages to all  $P_j \in \Gamma_j$
- 3: Every honest  $P_i$  accepts  $n$  forwardings from each  $P_j \in \Gamma_i$  and ignores the rest.  $P_i$  computes a set  $S_i$  of messages forwarded by at least  $(\delta - \alpha)n$  parties, and send  $S_i$  to all  $P_j \in \Gamma_i$

# Leader Selection: Protocol

## Protocol: Leader Selection w/ input $r$

- 1: Every honest  $P_i$  sends  $m_i = (i, \sigma_i(r))$  to all  $P_j \in \Gamma_i$
- 2: Every honest  $P_i$  forwards valid messages to all  $P_j \in \Gamma_j$
- 3: Every honest  $P_i$  accepts  $n$  forwardings from each  $P_j \in \Gamma_i$  and ignores the rest.  $P_i$  computes a set  $S_i$  of messages forwarded by at least  $(\delta - \alpha)n$  parties, and send  $S_i$  to all  $P_j \in \Gamma_i$
- 4: Every honest  $P_i$  receives a set  $S_j$  from every  $P_j \in \Gamma_i$ 
  - $P_i$  computes a set  $S_i^*$  of messages appear in at least  $(1 - \alpha)n$  received sets.
  - $P_i$  computes  $H(m_k)$  for every  $m_k \in S_i^*$ , and outputs  $P_l$  where  $l$  is the smallest id such that  $\forall m_k \in S_i^*, H(m_l) \leq H(m_k)$ .

# Leader Selection: Proof

Claim (All honest parties are accepted)

*In any iteration  $r$ , if  $P_i, P_j$  are any two honest participants, then  $m_j = (j, \text{Sig}_j(r)) \in S_i^*$  in **Step 4**.*

# Leader Selection: Proof

Claim (All honest parties are accepted)

*In any iteration  $r$ , if  $P_i, P_j$  are any two honest participants, then  $m_j = (j, \text{Sig}_j(r)) \in S_i^*$  in Step 4.*

- For all honest  $P_k$ : enough honest parties in  $\Gamma_k \cap \Gamma_j$  hence  $P_j \in S_k$  in Step 3.

# Leader Selection: Proof

Claim (All honest parties are accepted)

*In any iteration  $r$ , if  $P_i, P_j$  are any two honest participants, then  $m_j = (j, \text{Sig}_j(r)) \in S_i^*$  in **Step 4**.*

- For *all* honest  $P_k$ : enough honest parties in  $\Gamma_k \cap \Gamma_j$  hence  $P_j \in S_k$  in **Step 3**.
- $P_j$  is in at least  $(1 - \alpha)n$  sets sent to  $P_i$ , hence  $P_j \in S_i^*$ .

# Leader Selection: Proof

## Claim (Bound corrupt messages in $S_i$ )

*In any iteration  $r$ , if  $P_i$  is honest, and if  $\delta > 2\alpha$ , then  $S_i$  contains at most  $2n$  messages from corrupted participants.*

- # corrupted message forwarded to an honest  $P_i$  in Step 3:  
(blue from corrupted; red from honest)

$$\alpha n^2 + (1 - \alpha)\alpha n^2 = \alpha(2 - \alpha)n^2$$

- asdf



# Leader Selection: Proof

## Claim (Bound corrupt messages in $S_i$ )

*In any iteration  $r$ , if  $P_i$  is honest, and if  $\delta > 2\alpha$ , then  $S_i$  contains at most  $2n$  messages from corrupted participants.*

- # corrupted message forwarded to an honest  $P_i$  in **Step 3**:  
(*blue* from corrupted; *red* from honest)

$$\alpha n^2 + (1 - \alpha)\alpha n^2 = \alpha(2 - \alpha)n^2$$

- Since an accepted corrupted message “costs”  $(\delta - \alpha)n$  votes, the total # corrupted messages accepted by into  $S_i$  is at most:

$$\frac{\alpha(2 - \alpha)n^2}{(\delta - \alpha)n} \leq \frac{\alpha}{\delta - \alpha} 2n < 2n$$

# Leader Selection: Summary

## Lemma (Fairness of the protocol)

*If  $\delta > 2\alpha$ , there exists a three round Oblivious Leader Selection protocol with fairness  $1/(5n)$ . Further if  $\alpha < 1/2 - \epsilon$  for some positive constant  $\epsilon$ , then there exists a three round Oblivious Leader Selection protocol with constant fairness.*

(Let  $C$  be the set of honest parties.  $S^* = \cup_{P_i \in C} S_i^*$ .)

- asdf

# Leader Selection: Summary

## Lemma (Fairness of the protocol)

*If  $\delta > 2\alpha$ , there exists a three round Oblivious Leader Selection protocol with fairness  $1/(5n)$ . Further if  $\alpha < 1/2 - \epsilon$  for some positive constant  $\epsilon$ , then there exists a three round Oblivious Leader Selection protocol with constant fairness.*

(Let  $C$  be the set of honest parties.  $S^* = \cup_{P_i \in C} S_i^*$ .)

- A corrupted message accepted into  $S^*$  in **Step 4** has to appear in at least  $(1 - 2\alpha)n$  honest sets.

We bound # of corrupted messages in  $S^*$ :

$$\frac{2n|C|}{(1 - 2\alpha)n} = \frac{2}{1 - 2\alpha}|C|$$

# Leader Selection: Summary

## Lemma (Fairness of the protocol)

*If  $\delta > 2\alpha$ , there exists a three round Oblivious Leader Selection protocol with fairness  $1/(5n)$ . Further if  $\alpha < 1/2 - \epsilon$  for some positive constant  $\epsilon$ , then there exists a three round Oblivious Leader Selection protocol with constant fairness.*

(Let  $C$  be the set of honest parties.  $S^* = \cup_{P_i \in C} S_i^*$ .)

- Since  $1 - 2\alpha > 1/(2n)$ , the probability that an honest leader is elected is at least:

$$\frac{|C|}{|S^*|} \geq \frac{|C|}{|C| + \frac{2}{1/(2n)}|C|} = \frac{1}{1 + 4n} \geq \frac{1}{5n}$$

# Leader Selection: Summary

## Lemma (Fairness of the protocol)

*If  $\delta > 2\alpha$ , there exists a three round Oblivious Leader Selection protocol with fairness  $1/(5n)$ . Further if  $\alpha < 1/2 - \epsilon$  for some positive constant  $\epsilon$ , then there exists a three round Oblivious Leader Selection protocol with constant fairness.*

(Let  $C$  be the set of honest parties.  $S^* = \cup_{P_i \in C} S_i^*$ .)

- If  $\alpha < 1/2 - \epsilon$ , then  $1 - 2\alpha > 2\epsilon$  and the probability that an honest leader is elected is at least:

$$\frac{|C|}{|S^*|} \geq \frac{|C|}{|C| + \frac{2}{2\epsilon}|C|} = \frac{1}{1 + 1/\epsilon}$$

# Simplified Main Protocol

- We present only the simplified version
- refer to the thesis for the full version.

## Full Version:

- Honest parties **terminates** after reaching agreement

## Simplified Version:

- Honest parties **keeps the agreement** after reaching one

# Simplified Main Protocol

- We present only the simplified version
- refer to the thesis for the full version.

## Full Version:

- Honest parties **terminates** after reaching agreement
- Run until agreement is reached

## Simplified Version:

- Honest parties **keeps the agreement** after reaching one
- Run predetermined rounds and stop

# Simplified Main Protocol

- We present only the simplified version
- refer to the thesis for the full version.

## Full Version:

- Honest parties **terminates** after reaching agreement
- Run until agreement is reached
- **Agreement guaranteed** after termination

## Simplified Version:

- Honest parties **keeps the agreement** after reaching one
- Run predetermined rounds and stop
- **May fail** with negligible chance



# Main Protocol

Protocol: Simplified Main w/ input  $r \leftarrow 0$  and  $v_i$

- 1: Every honest  $P_i$  sends a random bit  $b \xleftarrow{R} \{0, 1\}$  to all  $P_j \in \Gamma_i$ , and then forwards received bits with ids to all  $P_j \in \Gamma_i$ .
- 2: Every honest  $P_i$  counts the forwarded bits:  $\forall j$ , if exists a unique value  $b_j^*$  with at least  $(\delta - \alpha)n$  votes, then set  $b_{ij} \leftarrow b_j^*$ . Otherwise, set  $b_{ij} \leftarrow 0$

# Main Protocol

Protocol: Simplified Main w/ input  $r \leftarrow 0$  and  $v_i$

- 1: Every honest  $P_i$  sends a random bit  $b \xleftarrow{R} \{0, 1\}$  to all  $P_j \in \Gamma_i$ , and then forwards received bits with ids to all  $P_j \in \Gamma_i$ .
- 2: Every honest  $P_i$  counts the forwarded bits:  $\forall j$ , if exists a unique value  $b_j^*$  with at least  $(\delta - \alpha)n$  votes, then set  $b_{ij} \leftarrow b_j^*$ . Otherwise, set  $b_{ij} \leftarrow 0$ .
- 3: Every honest  $P_i$  runs Leader Selection with  $r$ , and outputs  $P_{i_r}$ .

# Main Protocol

Protocol: Simplified Main w/ input  $r \leftarrow 0$  and  $v_i$

- 1: Every honest  $P_i$  sends a random bit  $b \xleftarrow{R} \{0, 1\}$  to all  $P_j \in \Gamma_i$ , and then forwards received bits with ids to all  $P_j \in \Gamma_i$ .
- 2: Every honest  $P_i$  counts the forwarded bits:  $\forall j$ , if exists a unique value  $b_j^*$  with at least  $(\delta - \alpha)n$  votes, then set  $b_{ij} \leftarrow b_j^*$ . Otherwise, set  $b_{ij} \leftarrow 0$ .
- 3: Every honest  $P_i$  runs Leader Selection with  $r$ , and outputs  $P_{l_i}$ .
- 4: Every honest  $P_i$  runs Graded Broadcast as the dealer with message  $v_i$ .  $P_i$  outputs  $(v_j, g_j)$  for every  $P_j \in \Gamma_i$ .
  - if at least  $(1 - \alpha)n$  1s have grades 1, then set  $v_i \leftarrow 1$ .
  - if at least  $(1 - \alpha)n$  0s have grades 1, then set  $v_i \leftarrow 0$ .
  - else set  $v_i \leftarrow b_{i,l_i}$ .
- 5: Increment  $r$  and *repeat*

# Main Protocol: Proof

## Claim (Honest random bit reaches all)

If  $P_i$  is honest and samples  $b_i \xleftarrow{R} \{0, 1\}$  in **Step 1**, then every honest  $P_j$  sets  $b_{ji} \leftarrow b_i$  in **Step 2**.

# Main Protocol: Proof

## Claim (Honest random bit reaches all)

If  $P_i$  is honest and samples  $b_i \xleftarrow{R} \{0, 1\}$  in **Step 1**, then every honest  $P_j$  sets  $b_{ji} \leftarrow b_i$  in **Step 2**.

- enough honest parties in  $\Gamma_i \cap \Gamma_j$  for all honest  $P_j$ , hence  $b_i$  has enough votes
- at most  $\alpha n < (\delta - \alpha)n$  votes forwarded by corrupted parties, hence  $b_i$  is unique.

# Main Protocol: Proof

## Claim (Agreement remains agreement)

*If all honest  $P_i$  start with  $v_i = v$  at **Step 1**, then they still have  $v_i = v$  at **Step 5**.*

## Claim (Honest leader leads to agreement)

*If all honest  $P_i$  outputs the same  $P_l$  in **Step 3** (Leader Selection), and  $P_l$  is honest, then with probability  $1/2$  they reach agreement.*

- Note if an honest  $P_i$  sets  $v_i \leftarrow 1$  in **Step 4**, then every other honest  $P_j$  either also sets  $v_j \leftarrow 1$  or sets  $v_j \leftarrow b_{jl_j}$

# Main Protocol: Proof

## Claim (Agreement remains agreement)

*If all honest  $P_i$  start with  $v_i = v$  at **Step 1**, then they still have  $v_i = v$  at **Step 5**.*

## Claim (Honest leader leads to agreement)

*If all honest  $P_i$  outputs the same  $P_l$  in **Step 3** (Leader Selection), and  $P_l$  is honest, then with probability  $1/2$  they reach agreement.*

- asdf

# Road Map

- Byzantine Agreement implies Broadcast
- Broadcast is impossible for  $\alpha \geq 1/2$
- Broadcast is impossible for  $\delta \leq 2\alpha$



# Broadcast: Definition

## Definition (Broadcast)

For  $N$  parties  $P_1, \dots, P_N$ , one of which is a distinguished dealer  $P_d \in S$  holding a message  $m$ , the following conditions must hold after the protocol:

- (Agreement) Every honest participant  $P_i$  outputs the same  $m^*$ , for some  $m^*$ .
- (Validity) If the  $P_d$  is honest, then  $m^* = m$ .

Note: if  $P_i \notin \Gamma_d$ ,  $P_i$  enters the protocol with the player id  $P_d$ , but not its public key  $Pk_d$ .

# Broadcast: Protocol

## Protocol: Broadcast

- 1: The dealer  $P_d$  runs a Graded Broadcast protocol with message  $m \in \{0, 1\}$ . Every honest  $P_i \in \Gamma_d$  outputs  $(m_i, g_i)$ .
- 2: For every honest  $P_i \in \Gamma_d$ :
  - if  $g_i = 1$ , then send  $m_i$  to all  $P_j \in \Gamma_i$  and set  $v_i \leftarrow m_i$ .
  - else, set  $v_i \leftarrow 0$
- 3: For every honest  $P_i \notin \Gamma_d$ :
  - if  $P_i$  receives at least  $(\delta - \alpha)n$  messages of a unique message  $m$ , then set  $v_i \leftarrow m_i$
  - Otherwise, set  $v_i \leftarrow 0$ .

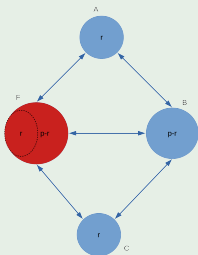
# Broadcast: Protocol

## Protocol: Broadcast

- 1: The dealer  $P_d$  runs a Graded Broadcast protocol with message  $m \in \{0, 1\}$ . Every honest  $P_i \in \Gamma_d$  outputs  $(m_i, g_i)$ .
- 2: For every honest  $P_i \in \Gamma_d$ :
  - if  $g_i = 1$ , then send  $m_i$  to all  $P_j \in \Gamma_i$  and set  $v_i \leftarrow m_i$ .
  - else, set  $v_i \leftarrow 0$
- 3: For every honest  $P_i \notin \Gamma_d$ :
  - if  $P_i$  receives at least  $(\delta - \alpha)n$  messages of a unique message  $m$ , then set  $v_i \leftarrow m$
  - Otherwise, set  $v_i \leftarrow 0$ .
- 4: Every honest  $P_i$  runs Byzantine Agreement with input  $v_i$ , and use its output  $v^*$  as the broadcast output.

$$\alpha \geq 1/2$$

### Example ( $\mathcal{C}_1$ )



**Figure:** A configuration  $\mathcal{C}_1$  with  $\alpha = 1/2$ , and  $\delta = (2p - r)/2p$ .

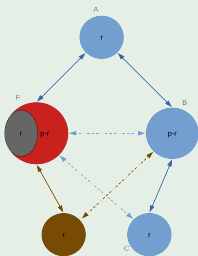
Claim ( $\mathcal{C}_1$  is valid)

$\mathcal{C}_1$  represents a valid configuration with  $\alpha = 1/2$  and any  $\delta = (2p - r)/(2p)$ .

- $\Gamma_A = A \cup B \cup F$
- $\Gamma_B = A \cup B \cup C \cup F'$
- $\Gamma_C = B \cup C \cup F$

$$\alpha \geq 1/2$$

### Example ( $\mathcal{C}_1$ )



**Figure:** An adversarial strategy for configuration  $\mathcal{C}_1$ .

### Claim ( $\mathcal{C}_1$ is impossible)

Let  $P_d \in C$  be some honest participant in group  $C$  with message  $m \in \{0, 1\}$ . We now claim that broadcast is impossible for  $P_d$  in  $\mathcal{C}_1$ .

(Adversary strategy:)

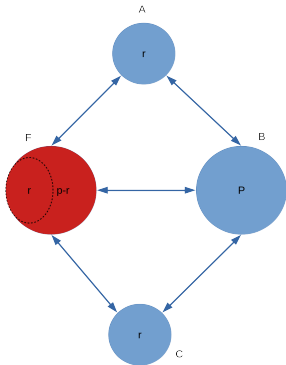
- simulates  $C'$  locally
- disables  $r$  parties in  $F$
- $F, C'$  ignores  $B, C$

# $\alpha \geq 1/2$ : Summary

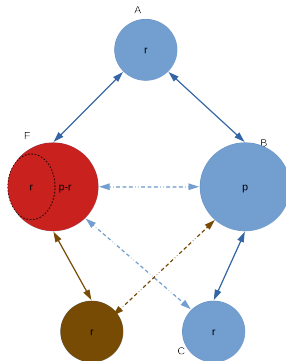
## Lemma ( $\alpha \geq 1/2$ )

*Assuming  $\alpha \geq 1/2$ , for any  $0 < \delta < 1$ , there does not exist a Broadcast protocol in our model.*

$$\delta \leq 2\alpha$$



(a) A configuration  $\mathcal{C}_2$  with  $\alpha = p/(2p + r) < 1/2$ , and  $\delta = 2\alpha$ .



(b) An adversarial strategy for configuration  $\mathcal{C}_2$ .

Figure: A counter example for the case of  $\alpha < 1/2, \delta \leq 2\alpha$ .

# $\delta \leq 2\alpha$ : Proof

Claim ( $\mathcal{C}_2$  is valid)

$\mathcal{C}_2$  represents a valid configuration with  $\alpha = p/(2p + r) < 1/2$  and  $\delta = 2\alpha$ .

Claim ( $\mathcal{C}_2$  is impossible)

Broadcast is impossible for  $P_d$  in configuration  $\mathcal{C}_2$ .

Lemma ( $\delta \leq 2\alpha$ )

Assuming  $\alpha < 1/2$  and  $\delta \leq 2\alpha$ , there does not exist a Broadcast protocol in our model



# Recap

## Theorem (Positive)

*If  $\delta > 2\alpha$ , there exists an expected  $O(n)$  round Byzantine Agreement protocol. Further if  $\alpha = 1/2 - \epsilon$  for any constant  $\epsilon$ , there exists an expected constant round Byzantine Agreement protocol in our model.*

# Recap

## Theorem (Positive)

*If  $\delta > 2\alpha$ , there exists an expected  $O(n)$  round Byzantine Agreement protocol. Further if  $\alpha = 1/2 - \epsilon$  for any constant  $\epsilon$ , there exists an expected constant round Byzantine Agreement protocol in our model.*

## Theorem (Negative)

*If  $\alpha \geq 1/2$  or  $\delta \leq 2\alpha$ , there does not exist a Byzantine Agreement protocol in our model*

# Future Work

## Question 1:

Note that our protocol runs in expected *constant* round if  $\alpha < 1/2 - \epsilon$  for any positive constant  $\epsilon$ . Does there exist an expected *constant* round protocol for  $\alpha < 1/2$ ?

# Future Work

## Question 1:

Note that our protocol runs in expected *constant* round if  $\alpha < 1/2 - \epsilon$  for any positive constant  $\epsilon$ . Does there exist an expected *constant* round protocol for  $\alpha < 1/2$ ?

## Question 2:

Note that our protocol somewhat defends against sybil attack, in that only corrupted parties connected to an honest party have effect. Can we make it fully resistant to sybil attack (possibly using *proof of work*)?



D. Dolev and H. R. Strong.

Authenticated Algorithms for Byzantine Agreement.

*SIAM Journal on Computing*, 12(4):656–666, November 1983.



Matthias Fitzi and Juan A. Garay.

Efficient player-optimal protocols for strong and differential consensus.

In *Proceedings of the twenty-second annual symposium on Principles of distributed computing - PODC '03*, page nil, - 2003.



Michael J. Fischer and Nancy A. Lynch.

A lower bound for the time to assure interactive consistency.

*Information Processing Letters*, 14(4):183–186, June 1982.



Pesech Feldman and Silvio Micali.

An optimal probabilistic protocol for synchronous byzantine agreement.

*SIAM Journal on Computing*, 26(4):873–933, 1997.



Juan A. Garay and Yoram Moses.

Fully Polynomial Byzantine Agreement for Processors in Rounds.

*SIAM J. Comput.*, 27(1):247–290, February 1998.



Jonathan Katz and Chiu-Yuen Koo.

On expected constant-round protocols for byzantine agreement.

In *Advances in Cryptology - CRYPTO*, Lecture Notes in Computer Science, pages 445–462. Springer Berlin Heidelberg, 2006.



Leslie Lamport, Robert Shostak, and Marshall Pease.

The Byzantine Generals Problem.

*ACM Transactions on Programming Languages and Systems*, 4(3):20.



S. Micali, M. Rabin, and S. Vadhan.

Verifiable random functions.

In *40th Annual Symposium on Foundations of Computer Science (Cat. No.99CB37039)*, page nil, - nil.



Silvio Micali and Vinod Vaikuntanathan.

Optimal and player-replaceable consensus with an honest majority.

2017.



M. Pease, R. Shostak, and L. Lamport.

Reaching agreement in the presence of faults.

*Journal of the ACM*, 27(2):228–234, April 1980.