# Smart TRB:

## An Incentive Compatible Consensus Protocol Utilizing Smart Contracts

Abhiram Kothapalli, Prof. Andrew Miller, Prof. Nikita Borisov

## ABSTRACT


- Previous work does not properly address incentive compatibility
- Propose a modified Dolev-Strong consensus algorithm that is incentive compatible for rational players
- Nodes follow protocol then report to a smart contract which dispenses rewards
- More efficient alternative to blockchain based consensus for smaller networks


## PROTOCOL DESCRIPTION

### Client Protocol

- Send request to node participating in consensus
- Later request output from any node
- Check that output contains leader signature

**Algorithm 1** Client Protocol

```
1: N ← smartContract_N
2: send input to q ∈ N                          ▷ wait after this step
3: i ← 0
4: result ← ∅
5: while result does not contain q_signature do
6:     result ← Nodei_outputs
7:     i ← i + 1
```

### Node Protocol

- Accepts requests from clients
- Runs a modified Dolev-Strong protocol for *K* rounds
  - *Predictable communication pattern:* Each node is required to send two messages to every other node each round. (acceptable message seq)
  - *Node status:* Each node maintains every other node as “good” or “bad”. A node labels nodes that don’t follow the acceptable message seq “bad”
- Nodes send final enemy report to smart contract after *K* rounds of Dolev-Strong are complete

**Algorithm 2** Node Protocol for arbitrary node $p$ numbered $p_i$

```
1: register with smart contract
2: outputs ← []
3: queue ← []
4: bad ← {}
5: in the background: queue ← queue + client_input
6: in the background: if client requests output, send outputs to client
7: while i < K do
8:     run Dolev-Strong with the following modifications:
9:     if i mod N = p_i then
10:        propose element from queue                ▷ p is the leader
11:    for all q ∈ N − {p} do
12:        if q ∈ bad then
13:            p does not forward messages to q
14:        if messages_q ∉ AcceptableMessageSequence then
15:            bad = bad + q
16:    outputs[i] = DSprotocol_output
17:    queue = queue − outputs[i]
18:    i ← i + 1
```

### Smart Contract Protocol

- Receives reports from all *N* nodes
- Receives total endowment *E*
- Set reward for each player *E/N*
- For every node *p, q* if *p’s* label of *q* is not equal to q’s label of *p*, *p’s* reward is reduced by some fixed constant θ
- Dispenses rewards to nodes

**Algorithm 3** Smart Contract Protocol

```
Ensure: receives report R_p for all p ∈ N
Ensure: receives endowment E > N^2 * θ
1: for all p ∈ N do
2:     reward_p ← E/N
3:     for all q ∈ N − {p} do
4:         if R_pq ≠ R_qp then
5:             reward_p ← reward_p − θ
6:     distribute reward_p to p
```

## BOUNDING REPORT COST, θ

- Under the assumption that node *p* will attempt to maximize its worst case utility

Because $p$ will attempt to maximize its worst case utility, we consider the case in which all $q \in N - \{p\}$ is playing the spiteful strategy, in which $q$ appears good to $p$ but $R_{pq} \neq R_{qp}$.

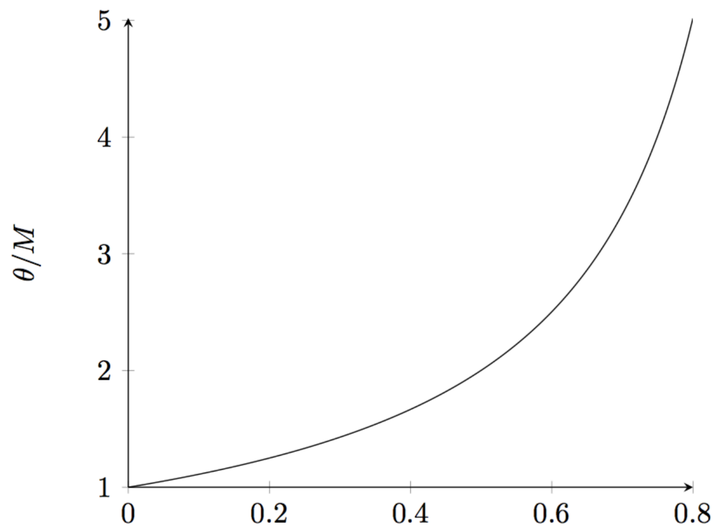
Strategy $\sigma_1$: Player $p$ sends messages to every $q \in N - \{p\}$

Strategy $\sigma_2$: Player $p$ randomly sends messages to $n\%$ of $q \in N - \{p\}$

In order for $p$ to always pick $\sigma_2$ we want:

$$\begin{aligned} \text{Cost}(\sigma_1) &\leq \text{Cost}(\sigma_2) \\ (N-1)*M + b*\theta &\leq (b + (N-1-b)*(n\%))*\theta + (N-1)*(1-n\%)*M \\ (N-1)*n\%*M &\leq (N-1-b)*n\%*\theta \\ \frac{N-1}{N-1-b}M &\leq \theta \end{aligned}$$

$b/N$ vs $\theta/M$ as $N \to \infty$




## CONCLUSION

- Studied the proper methodology to create, and rigorously prove the safety of BAR tolerant protocols
- Created and described our protocol from a game theory perspective to account for rational nodes
- Proved that our protocol is incentive compatible (therefore a Nash equilibrium)
- Argued why our protocol is an efficient alternative to fully blockchain based consensus algorithm

SCIENCE OF SECURITY
VIRTUAL ORGANIZATION
Funded by the National Security Agency.

INFORMATIONTRUST INSTITUTE