The Byzantine Generals Problem

- several generals, scattered in the area
- communication is through reliable links
- all connected to one another
- generals themselves not reliable
- each general has an initial value and eventually reaches a decision
- There is a bound $f$ on the number of generals that may be unreliable
The Problem

Design a protocol for the generals that guarantees:

**Agreement**  All nonfaulty generals decide on the *same value*

**Validity**  If all generals start with the *same initial value*, then this is the decision value

**Termination**  All nonfaulty generals eventually decide
**If Only Fail Stop Errors**

**FS-Agreement**  All generals *that decide*, decide on the same value

**FS-Validity**  If *all* generals start with the *same initial value*, then this is the decision value

**FS-Termination**  All nonfaulty generals eventually decide
Why is this Solvable?

(In view of our proof that there is no agreement in the presence of link failure...)

NYU G22.2631-001 – p.4
Digital Signatures

Each message sent is of the form

$$\langle i_1, \ldots, i_k; v \rangle$$

which abbreviates:

$$S_{i_1}(\cdots S_{i_{k-1}}(S_{i_k}(v))\cdots)$$

i.e., “$$i_1$$ signs: “$$i_2$$ signs: . . . “$$i_{k-1}$$ signs: “$$i_k$$ signs its value is $$v$$” ” . . . ”

When $$j$$ receives such a message, it knows the path the message travelled. It can then add its signature, and send it to others
Authenticated Synchronized Solution

- round \( k \) message is **good** if it is of the correct form and has \( k \) different signatures

- A **state** of a general consists of:
  - \( v_i \) an initial value;
  - \( V_i \) set of values, initially \( \{v_i\} \);
  - \( d_i \) a decision value, initially undefined;
  - \( r_i \) integers, initially \( 1 \);
  - \( Msg_i \) a set messages, initially \( \{v_i\} \) signed by \( i \);
  - \( U_i \) set of (received) messages, initially \( \emptyset \);

- at round \( k \): a general sends messages it receives w/o its signature; \( V_i \) maintains values received in good messages;

- at last round: if \( |V_i| = 1 \), then \( d_i = v \) s.t. \( V = \{v\} \). Else, \( d_i \) is some default value.
Authenticated BA: The Protocol

The state machine of each general is described by:

### (send)
- broadcast every message in $Msg_i$

### (receive)
- receive($u_1, \ldots, u_\ell$); store in $U_i$

### (local)
- $Msg_i := \emptyset$
- for every good $u = S_{i_1}(\ldots S_{i_{k-1}}(S_{i_{r_i}}(v))\ldots) \in U_i$
  - if $i_k \neq i$ for all $k = 1, \ldots, r_i$
    - $Msg_i := Msg_i \cup S_i(u)$
    - $V_i := V_i \cup \{v\}$
  - if $r_i = f + 1$ then $d_i := \text{decide}(V_i)$; halt
  - else $r_i := r_i + 1$

Where:

$$\text{decide}(V) = \begin{cases} \text{if } V = \{v\} & v \\ \hat{v} & \text{otherwise} \end{cases}$$
Correctness

Claim: If a value \( v \) is received in a (good) message sent by a non-faulty process, \( v \in V_i \) for all non-faulty processes \( i \) at the end of round \( f + 1 \),

Note: \( v \) is first received in a good message at round \( f + 1 \) ⇒ last process to sign it is non-faulty

Claim: At the end of \( f + 1 \) rounds, all non-faulty processes have the same \( V_i \).

Proof: If a non-faulty process \( i \) adds a value \( v \) to \( V_i \) before round \( f + 1 \), then it broadcasts it and by previous claim all non-faulty processes have that value in their \( V \)s. If \( i \) adds \( v \) to \( V_i \) at round \( f + 1 \), then it follows that it was sent by a good process so all non-faulty processes receive it and add it to their \( V \)s.
Correctness and Complexity

**Corollary:** The protocol is correct

**Proof:** Termination is guaranteed. Since all non-faulty processes have the same $V$ at the end, they all reach the same decision. If all processes started with $v$, then $v$ is the only value in the system and therefore at the end for every non-faulty $i$, $V_i = \{v\}$

**Time Complexity:** $f + 1$ rounds

**Communication Complexity:** In each round, each process sends $n$ messages, therefore, there are $O(n^2 f)$ messages

**Can we do better?** Each process only needs to send the first two values it sees (in good messages). Communication complexity is then $O(n^2)$
**Correctness Proof Outline**

**Claim:** If a correct process $i$ has upon termination $|V_i| \geq 2$, then so do all correct processes.

**Proof Outline:** Assume $v, v' \in V_i$ are first two values $i$ adds to $V_i$. Suppose, by way of contradiction, that for some correct $j$, $|V_j| < 2$. W.l.o.g, assume that $v \notin V_j$. If $i$ receives $v$ before phase $f + 1$, then it broadcasts it, and, since $j$ receives it, it adds $v$ to $V_j$. If $i$ receives $v$ at phase $f + 1$, it includes a signature of a correct process, and $j$ receives it too and includes $v$ in $V_j$. This contradicts that assumption that upon termination $v \notin V_j$. 
**Can We Do Even Better?**

**idea.** Divide the processes into $2f + 1$ active, and the rest passive. Actives run BA and after they decide (round $f + 2$) they send their (signed) decision to the passive

**correctness:** obvious

**time complexity:** $f + 2$ rounds

**comm. complexity:** $2(2f + 1)^2 + (2f + 1)n = \mathcal{O}(fn)$ msgs.

**Can last round be eliminated?** If passive processes *receive & not send* messages, they may receive values non-faulty don’t receive $\implies$ need a different decision to decide after $f + 1$ rounds!
Consider a passive process $p$

1. If $p$ receives 2 (or more) messages from each of $f + 1$ active processes, $p$ decides $\hat{v}$

2. ELSE, if $p$ has single $v$ signed by $f + 1$ (or more) active processes, (possibly in different messages) it decides $v$

3. ELSE (less than $f + 1$ active sent 2 values, and $f + 1$ support either 0 or more than 1 values), then $p$ decides $\hat{v}$

Note: This implies $f + 1$ rounds, and $2(2f + 1)n = O(fn)$ messages
**Proof of Correctness**

- all actives decide the same value (and if they all started with $v$, they all decide $v$)

- Assume active $a$ sees only $v$. Then all actives see only $v$. Thus, it cannot be the case that $f + 1$ active processes send two or more values, and it must be the case that $v$ (and only $v$) has $f + 1$ support $\iff$ all passive decide $v$.

- If *every* correct $a$ sees two or more values before phase $f + 1$, then every $p$ receives two values from $f + 1$ processes, thus decides $\hat{v}$

- It remains to consider the case that *every* correct $a$ sees two (or more) values by the end of phase $f + 1$, but some receive second value only at phase $f + 1$
Proof of Correctness (cont.)

every correct \(a\) sees two (or more) values by the end of phase \(f + 1\), but some receive second value only at phase \(f + 1\):

- some correct \(a\) has a single value at the end of round 1 (must be the initial value of every correct \(a\)) and every \(p\) has \(f + 1\) support for that value

- Consider the transmission of the second value received by a correct \(a\) at phase \(f + 1\) in a message \(m\). \(m\) must have come from a correct \(a'\), and has \(f + 1\) signatures \(\Rightarrow m\) is received by every \(p\)

- every \(p\) decides \(\hat{v}\)
**BA for Stopping Failures**

**Assumptions:** Synchronicity; Generals may have fail-stop faults

**Algorithm:** pretty much like the case of authentication:

1. **(send)** send($W_i$)
2. **(receive)** receive($u_1, \ldots, u_k$)
3. **(local)** $W_i := W_i \cup \{u_1, \ldots, u_k\}$

At round $f + 1$, add to **local**:

```
if |W_i| = 1 then $d_i = v_i$
else $d_i = \hat{v}$
```
**Proof of Correctness**

**Claim.** If all correct processes have $W_i^r = W$ after round $r \leq f$, they all have $W_i^{r+1} = W$

**Proof.** $W_i$'s are monotonically increasing. To increase $W_i$, $i$ has to receive a new $v_k \notin W_i^r$, which can happen only if $v_k \in W_j^r$ for some correct $j$. This is impossible since $W_i^r = W_j^r$ for all correct $j$'s.

**Claim.** If there is a round $r \leq f + 1$ without failures, then all correct processes have the same $W$ after it

**Proof.** At the end of round $r$, for every correct $i$

$$W_i^r = \bigcup_{j \text{ is correct at round } r} W_j^{r-1}$$

**Corollary.** Upon termination, all correct processes have the same $W$
Complexity

time. $f + 1$ rounds

messages $\leq n^2$ in each round, thus $O(n^2(f + 1))$

messages

bits. if values are at most $b$ bits, $O(bn^2(f + 1))$ bits

if only first two values sent. $f + 1$ rounds and

$O(2 \cdot n(f + 1))$ messages

If decision is some initial value: decide minimal $v \in V$, or,
for optimality, broadcast only minimal value observed
**Full Information Protocol**

- *Everybody* sends *everybody* the full (cycle-free) “gossip sequences”
- Message are of the form:
  
  \[ i_k \text{ said that } i_{k-1} \text{ said that } \ldots \text{ said that } i_1 \text{'s value is } v \]

  where the \( i_\ell \)'s are mutually distinct

- Gossip sequence is represented by a tree, with \( i_j \)'s labelling the edges

- A node whose path from the root is \( i_1, \ldots, i_k \) labelled by \( v \) indicates the receipt of the above message

- Each process maintains own tree set \( W \) of values seen
EIG Trees

- Initially tree consists of root is labelled with initial value
- At every round $r$, some of the nodes at depth $r$ are labelled with values, according to the messages the processes received. Since we assume stopping failure only, it is easy to see that $j$’s tree has a node $i_1, \ldots, i_k$ labelled by a non-null $v$ iff $i_k$’s tree has a node $i_1, \ldots, i_{k-1}$ labelled by $v$ and $i_k$ sent $i$ a message at round $k$. 
Correctness of ELG Protocol

All correct processes terminate with same $W$:

- Assume, by way of contradiction, there exists correct $i$ and $j$ such that upon termination $v \in W_i$ but $v \notin W_j$
- there exists a node $i_1, \ldots, i_k$ in $i$’s tree that is labelled by $v$ such that $i \neq i_\ell$ for every $\ell = 1, \ldots, k$
- if $k \leq f$, then at round $k + 1$, $i$ broadcasts the appropriate message (prefixed by “$i$ said”), so that every correct process labels its $i_1, \ldots, i_k, i$ node with $v$
- if $k = f + 1$, then there is some $\ell \leq k + 1$ such that $i_\ell$ is a correct process, and thus all the correct processes label their $i_1, \ldots, i_\ell$ by $v$
- in either case, $v \in W_j$, contradicting assumption
Consider a synchronous BA system with a (single) known sender and two possible values. Present an algorithm that achieves BA within $f + 3$ phases using at most $f(2 \cdot f + n)$ messages. You may assume digital signatures. (Hint: Not all actives run the same protocol.)

- State your algorithm clearly
- Prove all its properties:
  1. All correct processes reach the same decision
  2. If sender is correct then all decide on its value
  3. The complexity requirements are met
All the guidelines of the previous assignment apply (you *can* work in groups, you have to give full citation to any material you used, etc.)

Due: March 24th

*No extensions* will be granted

This is not a simple assignment; you may want to start thinking about it earlier rather than later