ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Lecture 8: Round and Communication Complexity of Consensus

CS 539 / ECE 526

Distributed Algorithms

Recall Dolev-Strong

- Lock-step synchronous, authenticated (i.e., use signatures) **deterministic** broadcast
  - Byzantine faults $f < n$
  - $f+1$ rounds
  - $O(n^2)$ messages
  - $O(n^2 f|\sigma|)$ bits

- Today: lower bounds on round and communication complexity of deterministic broadcast and agreement protocols

Outline

- Round complexity lower bound

- Communication complexity lower bound

Round Lower Bound

- No deterministic protocol can solve broadcast or agreement with f crash faults in f rounds
  [Fischer-Lynch,1982] [Dolev-Strong,1983] [Aguilera-Toueg,1999]

- Easier problem → stronger negative result
  - Binary, lockstep, crash → holds for harder models

Configurations

- Union of the states of all parties

- A protocol execution (in lockstep rounds) is an evolution of configurations, one per round
  - $C_0 \rightarrow C_1 \rightarrow C_2 \ldots$

Valency

- A config C is **0–valent**, if in all configs reachable from C, honest parties decide 0
  - No matter what happens from now on, decide 0
- A config C is **1–valent,** if ……, all decide 1
- **Univalent** = 0-valent or 1-valent
- **Bivalent** = not univalent

Valency Examples

- In broadcast, sender has input 1
  - Is this initial configuration univalent or bivalent?
- In agreement, every party has input 1
  - Is this initial configuration univalent or bivalent?
- Note that univalent ≠ some party decided
- In the deterministic and crash model, after f parties crash, the config becomes univalent

Intuition of the Proof

- Step 1: there exists an initial bivalent config
- Step 2: a new crash can maintain bivalency

- Thus, config can be bivalent after f crashes in f rounds (one crash per round)

Lemma 1: Initial Bivalency

- There exists an initial bivalent configuration
- For broadcast: when sender has input $\neq \perp$
- For agreement:
  - Suppose every initial config is univalent
  - C^i_0: first i parties have 0 and rest have 1
  - (1-val) C^0_0, C^1_0, C^2_0, ……, C^{i-1}_0, C^i_0, ……, C^n_0 (0-val)
  - $\exists$ i such that 1-valent C^{i-1}_0 and 0-valent C^i_0
    - Only i can tell the difference
  - What if i crashes ??? C^{i-1}_0 and C^i_0 become equivalent

Lemma 2: Maintains Bivalency

- There exists a bivalent C_{f-1} (after round f-1)
  - Base case: ∃ bivalent C_0
  - Inductive step: ∃ bivalent C_{k-1} → ∃ bivalent C_k
  - Suppose for contradiction every C_k is univalent
  - C^*_k = fail-free evolution of C_{k-1}. WLOG 0-valent.
  - C_{k-1} is bivalent → ∃ 1-valent C^{**}_k
    - Not fail-free. Suppose party p crashes in round k, without sending msg to $\{j_1, j_2, \ldots, j_m\}$ $(0 \leq m \leq n)$

Lemma 2 Proof Cont'd

- C^*_k = fail-free evolution of C_{k-1}. WLOG 0-valent.
- C^{**}_k = p crashes without sending msgs to $\{j_1, j_2, \ldots, j_m\}$
  - 1-valent
- C^i_k = p crashes without sending msgs to $\{j_1, j_2, \ldots, j_i\}$ $(0 \leq i \leq m)$
- (0-val) $C^0_k, C^1_k, C^2_k, \ldots\ldots, C^{i-1}_k, C^i_k, \ldots\ldots, C^m_k$ (1-val)
- $\exists$ i such that 0-valent C^{i-1}_k and 1-valent C^i_k
  - Only j_i can tell the difference
  - What if j_i crashes ???

Lemma 3: Final Disagreement

- Lemma 3: A bivalent configuration C_{f-1} leads to safety violation for any f-round protocol
  - Proof: Same as Lemma 2 except that j_i (the only one who can tell the difference) does not have a chance to tell others

- QED. f+1 rounds are needed for deterministic broadcast and agreement protocols.

What Can We Do?

- Optimize good case
  - E.g., t+2 rounds where t is actual # faults
  - E.g., constant rounds under a good leader

- Amortization: usually use round robin leaders

- Randomization: usually use random leaders

Outline

- Round complexity lower bound

- Communication complexity lower bound

Communication Complexity

- Different for crash vs. Byzantine
- For crash faults:
  - Trivial lower bound: $\Omega(n)$ messages and bits
  - Upper bound (best known algorithm): $O(n)$ messages and $O(n)$ bits [Galil-Mayer-Yung, 1995]

Communication Complexity

- Different for crash vs. Byzantine
- For crash faults: $\Theta(n)$ msgs and bits
- For Byzantine faults:
  - Will not count msgs sent by Byzantine parties
  - Lower bound: $\Omega(n+f^2)$ messages for any deterministic protocol [Dolev-Resichuk, 1985]
    - If $f = \Theta(n)$, $\Omega(n^2)$ msgs, met by Dolev-Strong

Dolev-Reischuk Lower Bound

- No deterministic protocol can solve Byzantine agreement with f faults in $f^2/4$ messages

- Easier problem → stronger negative result
  - Binary, lockstep → holds for harder models

Dolev-Reischuk Proof

- Suppose an algorithm uses < $(f/2)^2$ msgs
- Scenario S1:
  - Sender is honest and sends $v \neq \perp$
  - Let B be an arbitrary set of f/2 Byzantine parties
    - Parties in B do not send msgs to each other
    - Each party in B ignores the first f/2 msgs to it
  - Remaining parties (denoted A) output v (validity)
  - $\exists$ p $\in$ B that receives < f/2 msgs (pigeon hole)
  - Let A(p) be parties in A who send p msgs
    - Most likely, sender $\in$ A(p), but not important
  - We have $|A(p)| < f/2$

Dolev-Reischuk Proof

- Scenario S1:
  - Sender is honest and sends $v \neq \perp$, $|B| = f/2$ Byzantine
    - B do not send msgs to each other & ignores first f/2 msgs
  - Remaining A outputs v
  - $\exists\, p \in B$ such that $|A(p)| < f/2$
- Scenario S2:
  - A(p) and B \ p Byzantine (at most f/2+f/2)
  - B \ p behave like in S1 and ignore msgs from p
  - A(p) does not send msg to p, behave honestly otherwise
  - p receives no msg, **output** $\perp$ ← Safety violation! →
- A \ A(p) cannot distinguish S1 and S2, **output v**
  - B \ p, p, A(p) all behave the same towards A \ A(p)

Bit Complexity

- $\Omega(n+f^2)$ msg lower bound implies $\Omega(n+f^2)$ bits
  - For $f < n/3$, $O(n^2)$ bits achieved [Berman et al. 1992]
  - For $f < n/2$, $O(n^2|\sigma|)$ bits achieved [Momose-Ren, 2020]
  - For $f \geq n/2$, $O(n^2|\sigma|)$ bits achieved using heavy and non-standard cryptographic tools
- Some open questions remain, most notably the gap between $\Omega(n^2)$ and $O(n^2|\sigma|)$

What Can We Do?

- Good-case, e.g., O(nt) where t is actual # fault?
  – Very recent direction
- Randomization
  – Often sample a committee
- Amortization
  – Recent practical replication protocols go this route
  – But theoretically sound solutions only very recently

Summary

- Round and communication lower bounds for deterministic broadcast/agreement protocols
- Optimal round complexity: $f+1$
  - Achieved by Dolev-Strong
- Optimal msg complexity for Byzantine: $\Omega(n+f^2)$
  - Achieved by Dolev-Strong when $f = \Theta(n)$
- Optimal bit complexity: $\Theta(n^2)$ for $f < n/3$, gaps remain for other settings