DISTRIBUTED ALGORITHMS 2014

SEMINAR

Anastasija Efremovska and István Haller

Vrije Universiteit Amsterdam

27 November, 2014
Exercises for today:

- 11.4 (chapter Synchronous Networks)
- 13.7 - 13.12 (chapter Byzantine Generals problem)
Suppose that performing internal events at the end of a pulse takes time, and that an upper bound $e$ is known for this processing time. Explain how the synchronizer for bounded delay networks, with local clocks that have $\rho$ bounded drift and precision $\delta$, needs to be adapted. Argue that your adapted synchronizer is correct.
Reminder: Synchronous Networks

- Local clocks are $\rho$ bounded to real time
- Local clocks are $\delta$ synchronized to each other
- A message sent at local time $(i - 1)(\rho^2 \times \delta + \rho d_{max})$ is always received before local time of another process reaches $i(\rho^2 \times \delta + \rho d_{max})$
- If a received message needs to be processed, next pulse cannot start immediately
**Exercise 13.7**

**Lamport-Shostak-Pease**: Let \( N = 5 \) and \( k = 1 \), and let the general \( g \) be Byzantine. Suppose that in pulse 1, \( g \) sends the value 1 to two lieutenants, and the value 0 to the other two lieutenants. Give a computation of \( \text{Broadcast}_g(5, 1) \) (including a definition of the *majority* function) such that all lieutenants decide for 0.
REMINDER: LAMPORT-SHOSTAK-PEASE

- General $g$ performs $\text{Broadcast}_g(N, m)$ with the proposed value as pulse 1
- When a lieutenant $p$ receives $\text{Broadcast}^S_q(N - i, k - i)$ in pulse $i + 1$:

**If $k - i > 0$**

- Cannot trust value, set up multiset for particular Broadcast
- Add received value to new multiset
- Send $\text{Broadcast}_p^{SUq}(N - i - 1, k - i - 1)$ to vote for matching multiset at other processes (pulse $i + 2$)
- Wait for votes of the type $\text{Broadcast}_r^{SUq}(N - i - 1, k - i - 1)$ from other processes $r$ to fill multiset (pulse $i + 2$)
- Missing votes in pulse $i + 2$ are replaced by 0
- Final decision is $\text{majority}(\text{multiset})$
REMINDER: LAMPORT-SHOSTAK-Pease
CONTINUED

If \( k - i = 0 \)
- Can trust value, immediately add to proper multiset

Why exclude \( k \) nodes?
- Broadcast from Byzantine process unsafe if other Byzantine processes still voting
- Try to eliminate all Byzantine processes
- Broadcast of correct process safe if majority is still correct \((K < \frac{N}{3})\)

- Final decision is the *majority* vote for the multiset generated for \( Broadcast_g(N, k) \)
- A good blog entry if you want to read up on the algorithm: http://marknelson.us/2007/07/23/byzantine/
Lamport-Shostak-Pease: Let $N = 7$ and $k = 2$, and let the general $g$ and one lieutenant be Byzantine. Give a computation of $\text{Broadcast}_g(7, 2)$ (and its subcalls) in which all correct lieutenants decide for $\text{majority}(0; 0; 0; 1; 1; 1)$. 
**Exercise 13.9**

*Lamport-Shostak-Pease*: Determine the worst-case message complexity of $\text{Broadcast}_g(N, k)$ for the correct processes.
Apply the Lamport-Shostak-Pease authentication algorithm to a complete network of five processes. Let three of the processes be Byzantine. Give a computation in which the two correct processes would decide for different values at the end of pulse 3, but decide for the same value in pulse 4.
Public key cryptography used to avoid forged messages

$S_p(\nu)$ is value $\nu$ encrypted with the secret key of $p$

Encrypting with secret key is equivalent to signing message

Signing makes sure that you cannot forge general

He may still be byzantine

When a process receives a valid message, it forwards it in the next pulse, adding its own signature to it

A message is valid if:

1. It contains $i$ process signatures if we are currently in pulse $i$
2. The first signature belongs to the general
3. All processes are different
4. All processes signed the same value

After pulse $k + 1$ every correct process has same set

Decide for $\nu$ if only $\nu$ is contained in set, 0 otherwise
Let $k \geq N - 1$. Explain why the Lamport-Shostak-Pease authentication algorithm can then be adapted by letting it already terminate at the end of pulse $N - 1$. 
At most one correct process.
Exercise 13.12

Determine the worst-case message complexity of the Lamport-Shostak-Pease authentication algorithm for the correct processes, taking into account the Dolev-Strong optimization.
REMINDER: DOLEV-STRONG OPTIMIZATION

- Only forward every value you received once.
- Maximum of two broadcasts for every process.