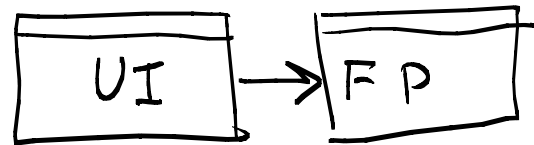


# Designing Distributed Algorithms

Note Title

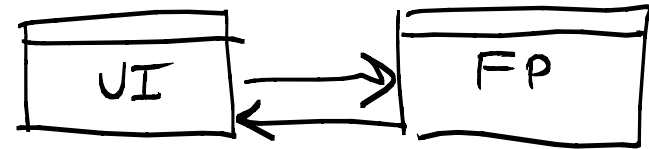
4/12/2007

## Quiz 4:



(a)

"UI calls FP"



(b)

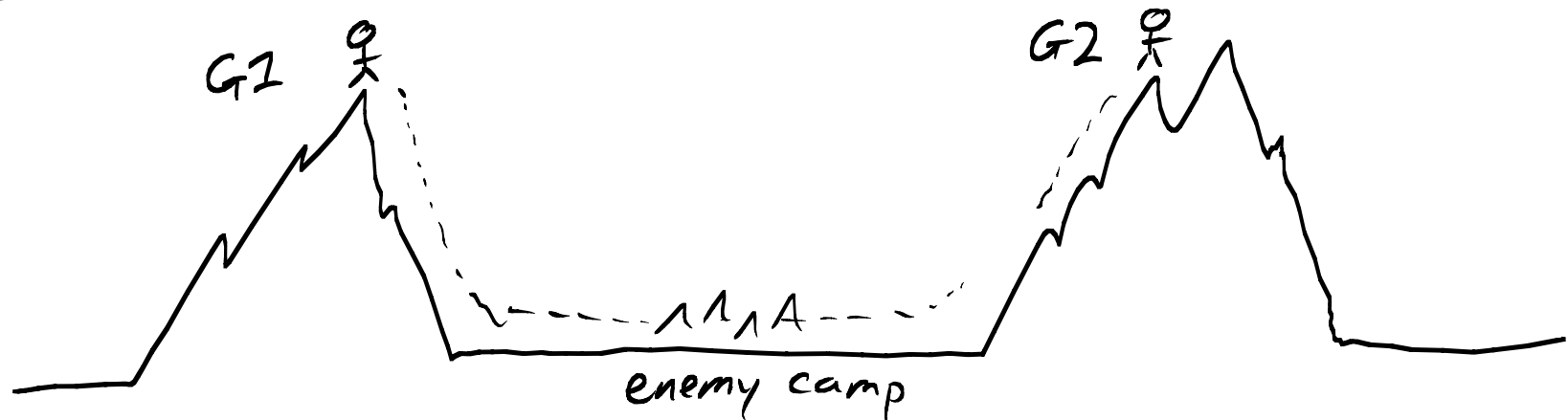
"UI + FP call each other"

A. Which of these two diagrams is better for regression testing? Why?

B. If something like (b) were chosen to support user feedback during a long operation (like fetching files from the network), what [modification] could be introduced to support multiple UIs?

\* Remember to put your name & lab section at the top.

## Distributed Algorithm Problem 1:

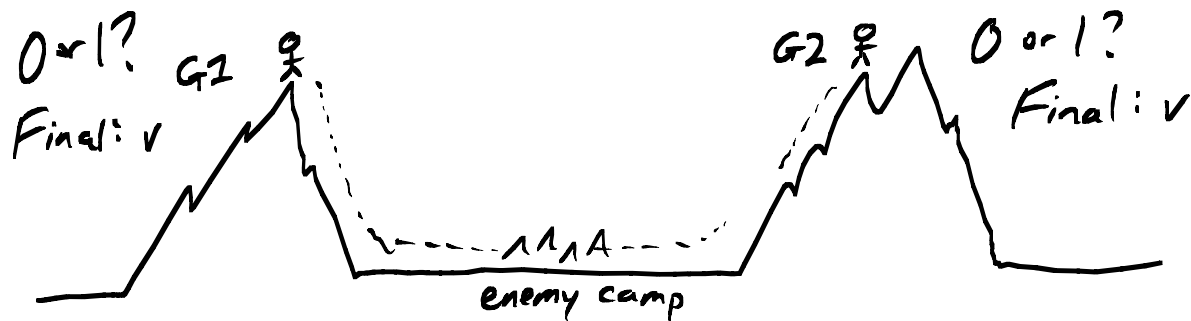


Two generals problem

- Each assess the situation (attack or not?) 1 or 0
- Communicate through messengers — can be captured
- At the end of the message exchange:

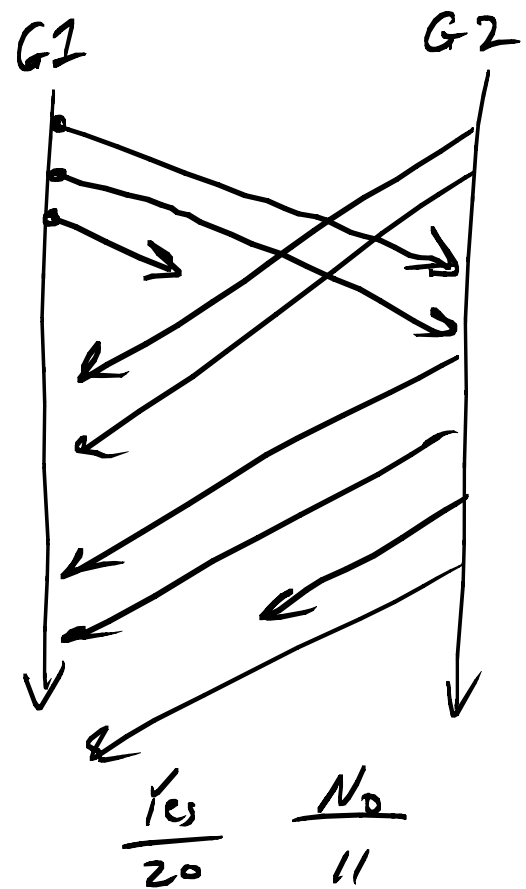
AGREEMENT: Both reach the same decision.

VALIDITY: The final decision must be among the initial assessments.



Two generals problem

- Each assess the situation (attack or not?) 1 or 0
- Communicate through messengers — can be captured
- At the end of the message exchange:
  - AGREEMENT: Both reach the same decision.
  - VALIDITY: The final decision must be among the initial assessments.

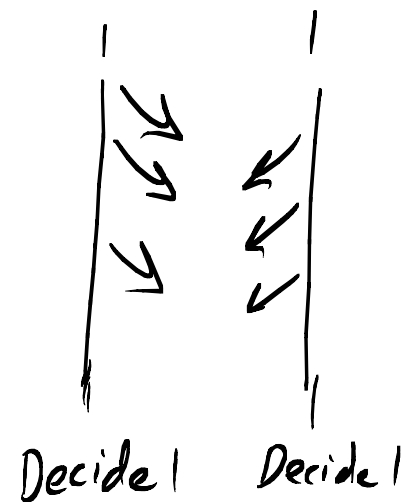
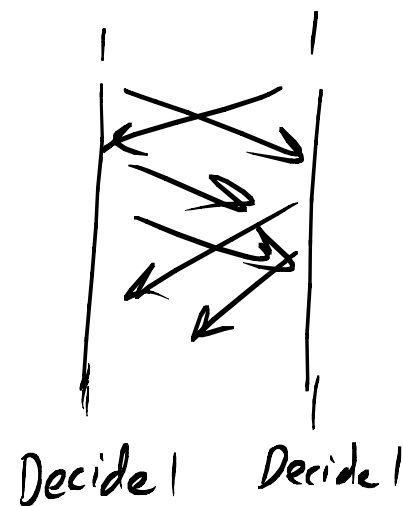
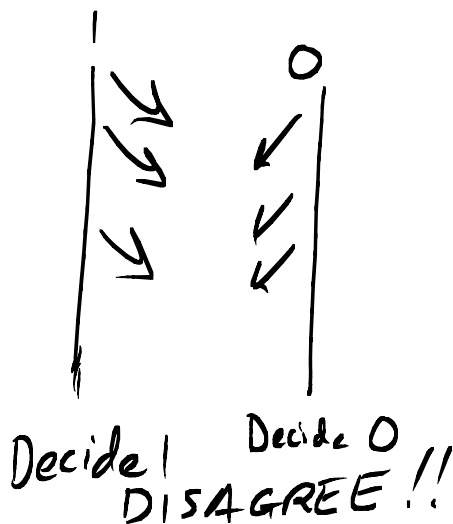
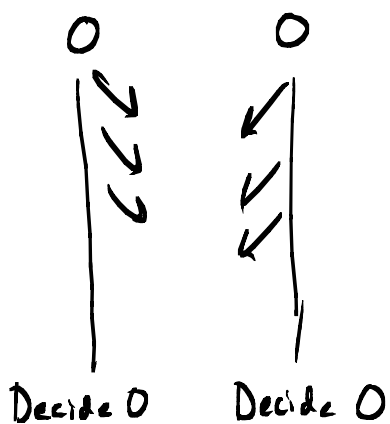
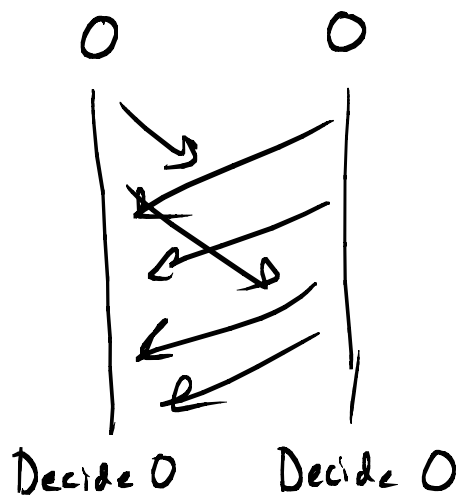


Ideas:

- ① Send lots of messages
- ② Put a sequence # on each message
- ③ Encrypt the messages — assume can tell (& discard) if a message is corrupted
- ④ Use timeouts to send again when not getting a response

Theorem: The 2-generals problem is unsolvable. There is no protocol.

Proof: Assume there exists some protocol.

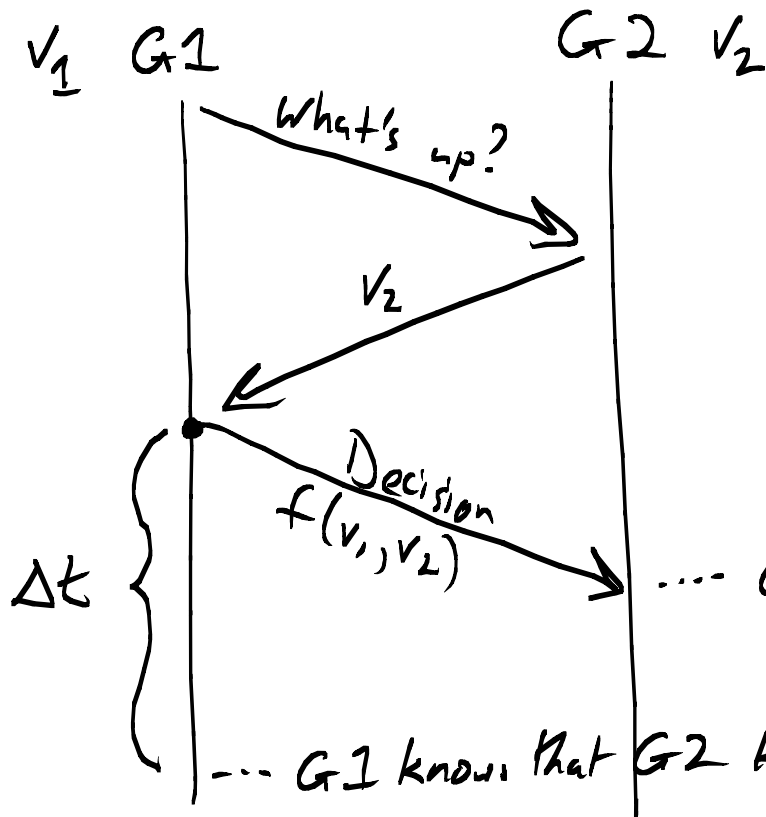


Issue: Completely asynchronous.

Don't know when other side has knowledge.

Assume: Stronger comm. model.

Time bound  $\Delta t$  on delivery.

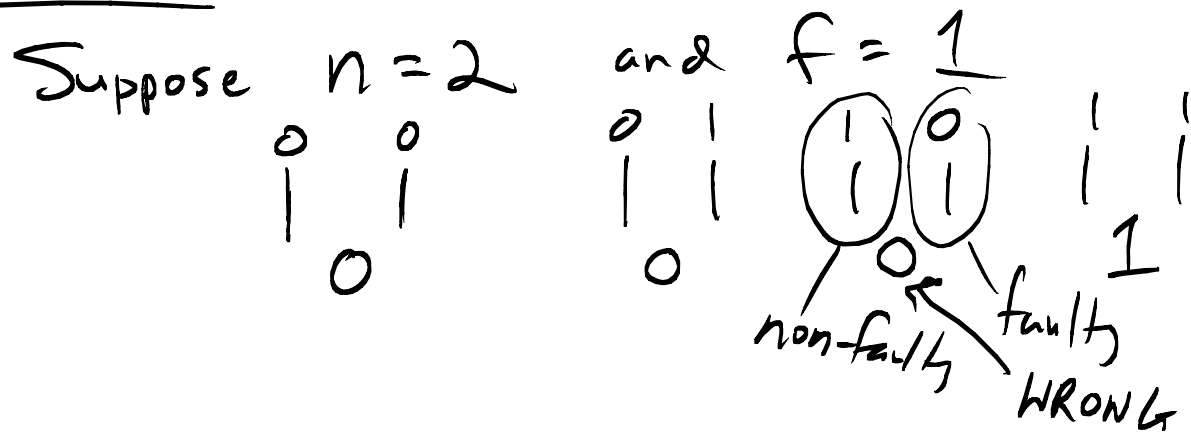


Common knowledge:  
"Everybody knows  
that everybody knows"

... G2 knows the decision  
... G1 knows that G2 knows the decision  
G2 knows that G1 knows that G2 knows.

## Problem 2: Byzantine Generals Problem

- An number  $n$  processes.
- Of these  $n$ , up to  $f \leq n$  may be faulty:
  - stop sending messages (crash failure)
  - skip over messages (omission failure)
  - can lie to disrupt the system (Byzantine failure)  
(can collude)



### Agreement:

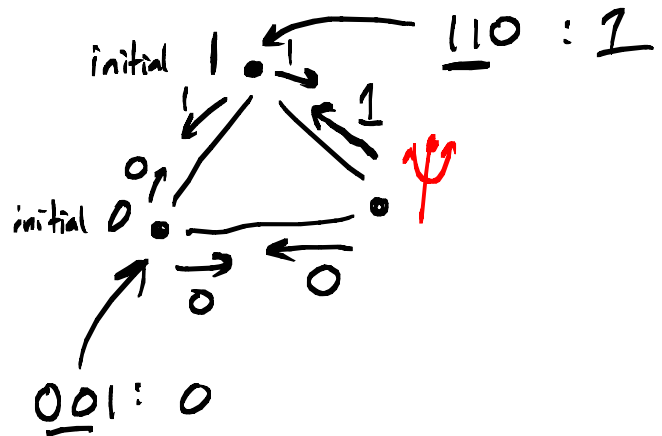
- All non-faulty agree.

### Validity:

- Final value was initial value of some  $n-f$  process.

What if  $n=3$  and  $f=1$  ?

Idea: Vote (Default 0)



Agreement:

- All non-faulty agree.

Validity:

- Final value was initial value of some  $n-f$  process.