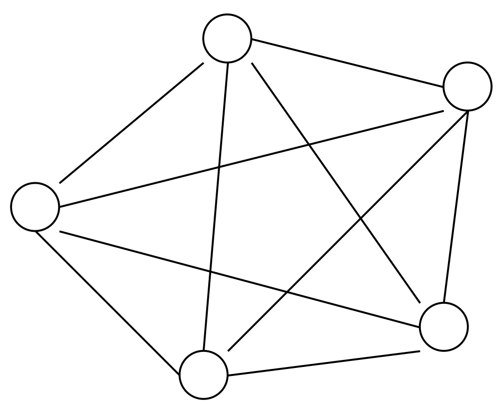
Byzantine Generals Problem

Statement of the problem

N generals have to agree about a plain of action: whether to ***attack*** or to ***retreat*** during a phase of the war.

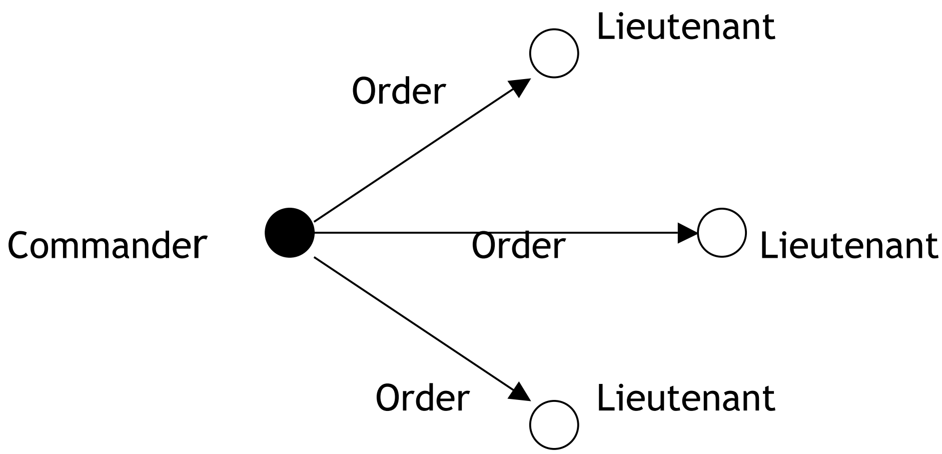


Some generals are traitors. Their actions can be modeled as Byzantine failures.

Synchronous system – message delays have upper bounds. The topology is completely connected.

How will they reach consensus?

Interactive Consistency Criteria



The roles will switch and the generals will take turns to broadcast their orders.

IC1. Every loyal lieutenant receives the same order from the commander.

IC2. If the commander is loyal, then every loyal lieutenant receives the order that the commander sends.

Communication using Oral messages

Messages are not corrupted in transit.

The absence/ loss of messages can be detected.

Receiver's / defaulter's identity is known.

Consensus using oral messages

The goal of OM(m) is to satisfy IC1 & IC2 in presence of m traitors and n generals.

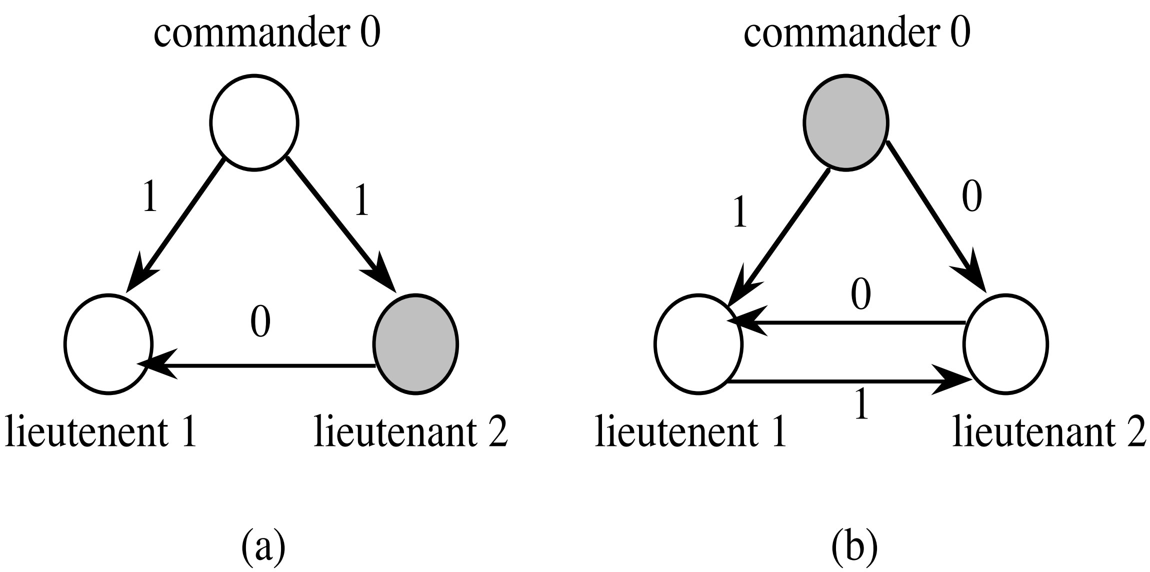
Review the easy case of m = 0. OM(0) is direct communication.

When m > 0, *indirect communication* is necessary. Each lieutenant will ask other lieutenants: *What order did you get from the commander?* Hopefully, this might resolve inconsistent orders by a traitor

An impossibility result

Using oral messages, no solution is possible if n≤3m.

Consider the case m=1



(a) Commander is loyal (b) Commander is a traitor

If you can prove the result for m = 1, then you can prove the general result by dividing all m traitors into one group.

The OM(m) algorithm

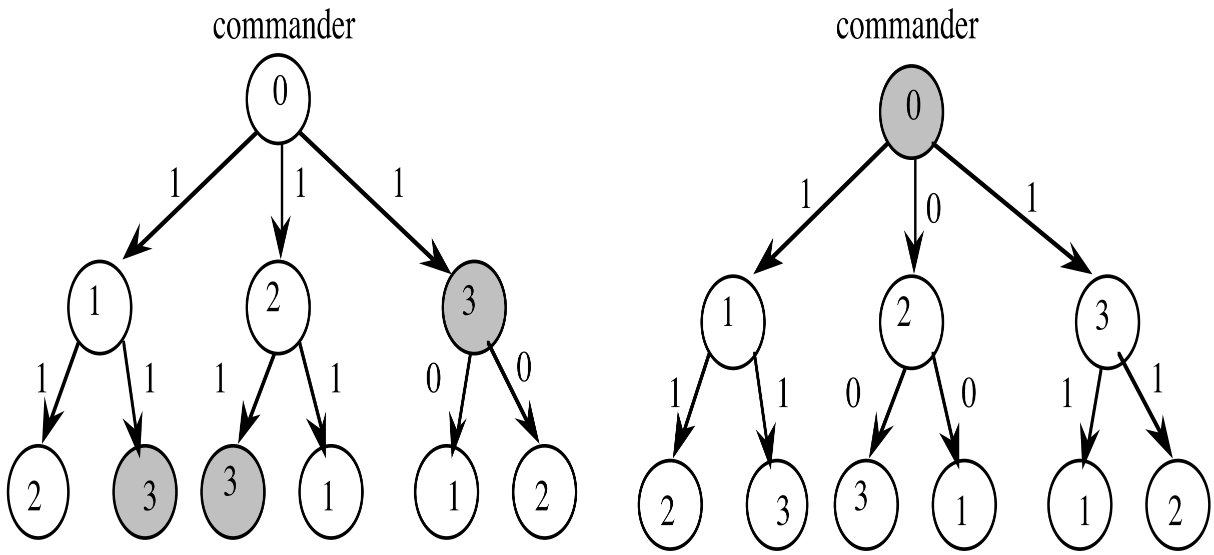
OM(0)

1. The commander **i** sends out a value **v** (0 or 1) to every lieutenant **j** **(j ≠ i)**, and each lieutenant **j** accepts it as the order from commander **i.**

OM(m)

1. The commander **i** sends out a value **v** (0 or 1) to every lieutenant **j** **(j ≠ i)**
2. If **m > 0**, then each lieutenant **j**, after receiving a value from the commander, initiates **OM(m-1)** Each lieutenant thus receives **(n-1)** values: a value *directly* received from the commander **i** and **(n-2)** values *indirectly* received orders from the **(n-2)** lieutenants when they executed **OM(m-1)**.
3. Each lieutenant chooses the *majority* of the **(n-1)** values received by it as the *order* from the commander **i.**

An illustration of OM(1)



(a)

(b)

Example with m=1 and n=4

The total number of messages required is

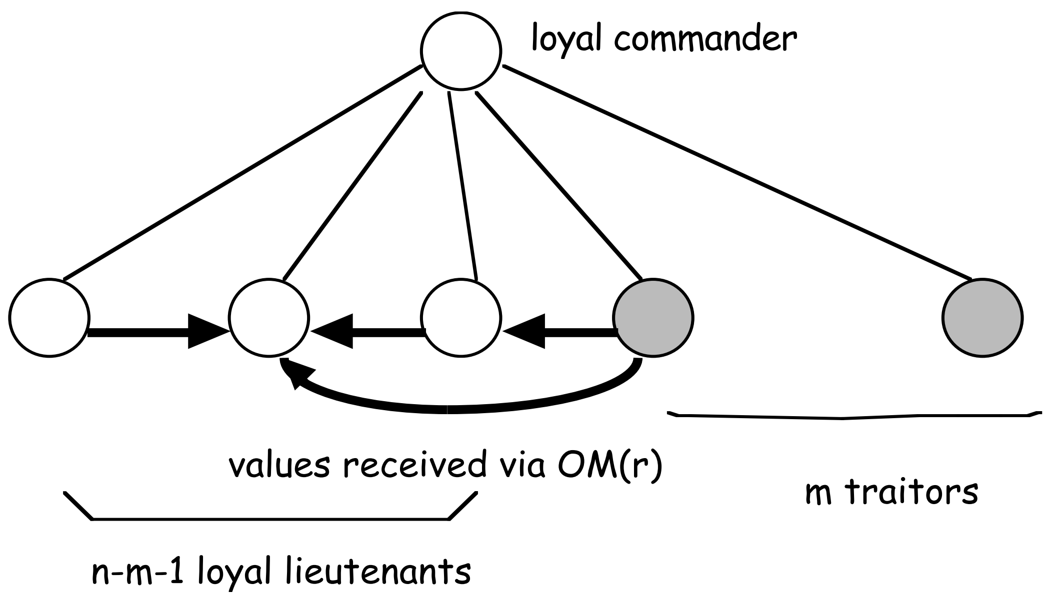
(n-1)(n-2)(n-3)… (n-m), i.e. O(n^{m})

Quite inefficient!

Study an example with m=2 and m=7.

Proof of the oral message algorithm

Lemma Let the *commander be loyal*, and **n > 2m+k**, where **m** = maximum number of traitors. Then **OM(k)** satisfies **IC2**.



Basis. The case k=0 is trivial

Inductive step. Let it hold for k=r. Show that it holds for k=r+1.

By assumption n >2m + r + 1. So n-1 > 2m + r > 2m.

The values received via OM(r) are good (induction hypothesis). So, a majority of the values received by the lieutenants are good.

Theorem. If **n > 3m** then **OM(m)** satisfies both **IC1** and **IC2**.

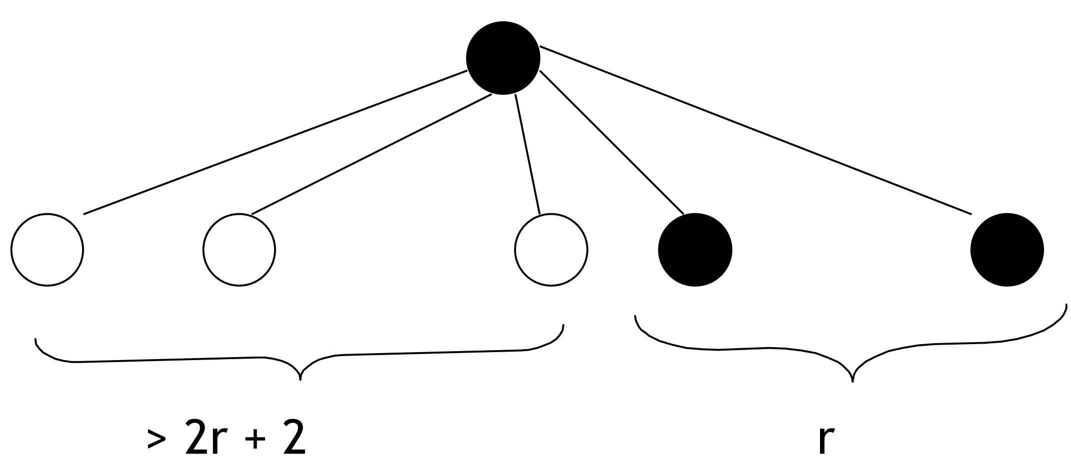
Basis. When **m = 0**, the theorem trivially holds.

Inductive Step. Let it hold for **m=r.** Show that it holds for m=**r+1**.

Substitute **k = m** in the lemma. Two cases:

Case 1. Commander is loyal. Then **OM(m)** satisfies **IC2**, and hence **IC1**.

Case 2. Commander is a traitor**.** There are more than 3r+3 traitors, and there are r+1 traitors.



Each loyal lieutenant **i** will receive the same order from every other loyal lieutenant **j** -it is the value that **j** received from the (traitor) commander.

By the *induction hypothesis*, each loyal lieutenant will receive identical orders from the **r** traitors. So any choice function (like majority) on the set will produce the same result.