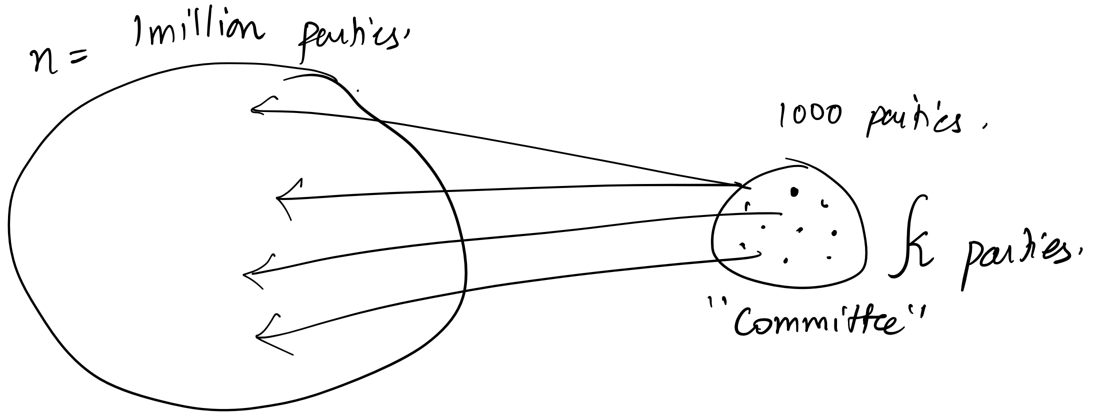
Dolev-Reischuk: Any deterministic protocol requires $O(n^2)$ communication

Idea: One party p that does not receive any message. P will disagree with other honest parties.

Q: Can we obtain a subquadratic communication protocol if we assume randomization?



"Every party to every other party": $O(n^2)$ $\downarrow$ 10^{12}

"Every party in the committee sends to every other party": $O(kn)$

$\Longrightarrow (k)$

(n)

Coin c. (256 bit) "Common coin".

$H(id_p, \underset{\Uparrow}{c}) \Longrightarrow$ if this value is "small enough," then the party is in the committee.

If we set k as the expected committee size, since every party is elected uniformly & independently at random, we can argue that the committee size $(1-\delta)k <$ committee size $< (1+\delta)k$. except with probability $e^{-\Omega(\delta k)}$. (Chernoff bound).

If we start off with $\frac{1}{3} - \epsilon$ Byzantine $\to k$.

$2^{-40} \Rightarrow -50$ $\quad \epsilon \to 0.05$

parties in the entire population, then the committee will have $< \frac{1}{3}$ Byzantine parties.

Protocol! Secure against a static adversary.

- Elect committee C of size k. (no communication)
- Run the agreement protocol within the committee. ($O(k^2)$ communication)
- All parties learn the output by communicating with sufficiently many committee members. ($O(nk)$ communication)

$$O(nk + k^2) = O(nk) =$$

Identities of committee members are public:

- An adversary can attack all committee members
- Bribery.

Adaptivity of the adversary.

Static: Adversary corrupts upto t parties at the beginning of the protocol.

Adaptive: Adversary corrupts parties during the execution.

- A party can be corrupted at the start of a round.

Mobile:

Bitcoin: Static ✓ or adaptively secure ✓?

"a miner useful now is not useful at a later point":

(A) ↳ I do not know whether I am "in the committee" until I mine PoW.

(Verifier)
(B) ↳ Once I do win, I can prove to the world, that I am the winner.

(C) ↳ An adversary cannot modify the contents of the block (even by corrupting me).

(Rand)
(D) ↳ Randomized / fair.

(B) - (C) - (D).

Verifiable Random function:

$VRF(\underline{sk_i}, r,) \longrightarrow o/p$:

r ↓ prev_hash / Genesis's + round

$VRF_{prove}(sk_i,) \rightarrow o/p$

$VRF_{verify}(\times , pk_i) \Rightarrow 1, 0$

num.

- ↳ Uniformly random.
- ↳ Verifiable.
- ↳ Unique

↳ Change secret keys fequently: Key evolving signatures.

(pk_i, sk_i^x) — Round x.

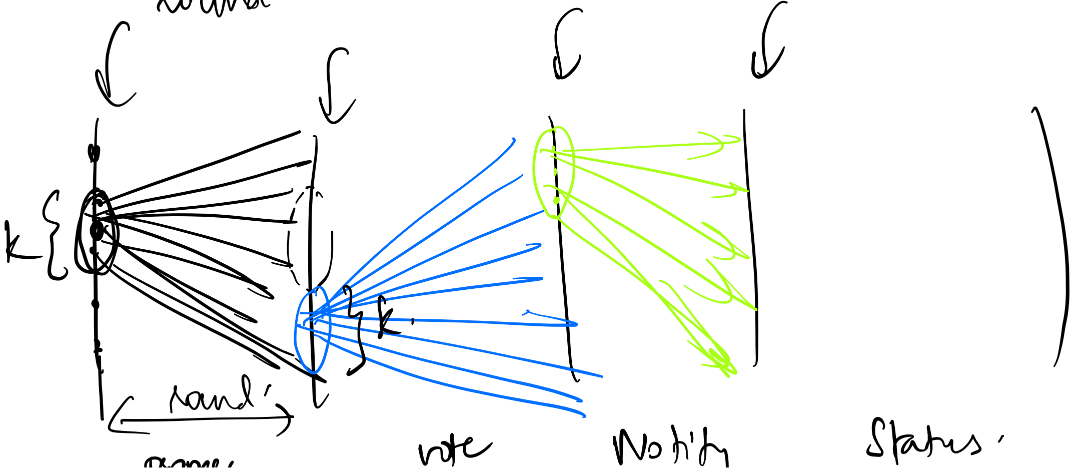
$\downarrow$ $\qquad\qquad$ $\downarrow$

(pk_i, sk_i^{x+1}) $\qquad$ $x+1$

- ↳ If I am elected, create a local msg/proposal.
- ↳ Evolve my key $sk_i^x \rightarrow sk_i^{x+1}$
- ↳ Delete sk_i^x.
- ↳ Send the proposal to everyone.

Player-replaceability:

- ↳ Use subquadratic-comm^n protocol fromearlier.
- ↳ Elect a different committee in each round.



purpose

$O(n \, \text{poly}(k))$,

Player-replaceability: Sub-quadratic comm^n + adaptive adversary.