The Triumph of Randomization

The Big Picture

Does randomization make for more powerful algorithms?
- Does randomization expand the class of problems solvable in polynomial time?
- Does randomization help compute problems fast in parallel in the PRAM model?

You tell me!

The Triumph of Randomization?

Well, at least for distributed computations!
- no deterministic 1-crash-resilient solution to Consensus
- \( f \)-resilient randomized solution to consensus (\( f < \frac{n}{2} \)) for crash failures
- randomized solution for Consensus exists even for Byzantine failures!

A simple randomized algorithm

M. Ben Or. “Another advantage of free choice: completely asynchronous agreement protocols” (PODC 1983, pp. 27-30)
- exponential number of operations per process
- BUT more practical protocols exist
  - down to \( O(n \log^2 n) \) expected operations/process
  - \( n - 1 \) resilient
The protocol's structure

An infinite repetition of asynchronous rounds

in round \( r \), each \( p \) only handles messages with timestamp \( r \).

each round has two phases

in the first, each \( p \) broadcasts an a-value which is a function of the b-values collected in the previous round (the first a-value is the input bit)

in the second, each \( p \) broadcasts a b-value which is a function of the collected a-values

 decide stutters

Ben Or's Algorithm

1: \( a_p := \text{input bit}; r := 1; \)
2: repeat forever
3: (phase 1)
4: send \((a_p, r)\) to all
5: Let \( A \) be the multiset of the first \( n-f \) a-values with timestamp \( r \) received
6: if \((\exists v \in (0, 1)) : \forall a \in A \cdot a = v\) then \( b_p := v \)
7: else \( b_p := 1 \)
8: (phase 2)
9: send \((b_p, r)\) to all
10: Let \( B \) be the multiset of the first \( n-f \) b-values with timestamp \( r \) received
11: if \((\exists v \in (0, 1)) : \forall b \in B \cdot b = v\) then decide\( (v) \); \( a_p := v \)
12: else if \((\exists b \in B : b \neq 1)\) then \( a_p := b \)
13: else \( a_p := \$ \) \{ \$ is chosen uniformly at random to be 0 or 1 \}
14: \( r := r + 1 \)

Validity

1: \( a_p := \text{input bit}; r := 1; \)
2: repeat forever
3: (phase 1)
4: send \((a_p, r)\) to all
5: Let \( A \) be the multiset of the first \( n-f \) a-values with timestamp \( r \) received
6: if \((\exists v \in (0, 1)) : \forall a \in A \cdot a = v\) then \( b_p := v \)
7: else \( b_p := 1 \)
8: (phase 2)
9: send \((b_p, r)\) to all
10: Let \( B \) be the multiset of the first \( n-f \) b-values with timestamp \( r \) received
11: if \((\exists v \in (0, 1)) : \forall b \in B \cdot b = v\) then decide\( (v) \); \( a_p := v \)
12: else if \((\exists b \in B : b \neq 1)\) then \( a_p := b \)
13: else \( a_p := \$ \) \{ \$ is chosen uniformly at random to be 0 or 1 \}
14: \( r := r + 1 \)

Validity

All identical inputs \( (i) \)

Each process sets a-value := \( i \) and broadcasts it to all

Since at most \( f \) faulty, every correct process receives at least \( n-f \) identical a-values in round 1

Every correct process sets b-value := \( i \) and broadcasts it to all

Again, every correct process receives at least \( n-f \) identical i b-values in round 1 and decides
A useful observation

For all $r$, either

- $b_{p,r} \in \{1, \perp\}$ for all $p$ or
- $b_{p,r} \in \{0, \perp\}$ for all $p$

A useful observation

For all $r$, either

- $b_{p,r} \in \{1, \perp\}$ for all $p$ or
- $b_{p,r} \in \{0, \perp\}$ for all $p$

Lemma

Let $B$ be the multiset of the first $n-\alpha$-values with timestamp $r$ received:

- $\exists (b,v) \in B; v \neq \perp$

Lemma

Let $A$ be the multiset of the first $n-\alpha$-values with timestamp $r$ received:

- $\exists (a,v) \in A; v \neq \perp$

Proof

By contradiction.

Suppose $p$ and $q$ at round $r$ such that $b_{p,r} = 0$ and $b_{q,r} = 1$

From lines 6,7 $p$ received $n-\beta$ distinct Os, $q$ received $n-\beta$ distinct Is. Then, $2(n-\beta) \leq n$, implying $n \leq 2\beta$.

Corollary

It is impossible that two processes $p$ and $q$ decide on different values at round $r$.

Agreement

Let $r$ be the first round in which a decision is made.

Let $p$ be a process that decides in $r$.

Let $r$ be the first round in which a decision is made.

Let $p$ be a process that decides in $r$.

By the Corollary, no other process can decide on a different value in $r$.

To decide, $p$ must have received $n-\beta$ from distinct processes.

Every other correct process has received $\gamma$ from at least $n-2\beta \geq 1$

By lines 11 and 12, every correct process sets its new $a$-value to for round $r+1$ to $\gamma$.

By the same argument used to prove Validity, every correct process that has not decided $\gamma$ in round $r$ will do so by the end of round $r+1$.
Termination I

> Remember that by Validity, if all (correct) processes propose the same value \(i\) in phase 1 of round \(r\), then every correct process decides \(i\) in round \(r\).

> The probability of all processes proposing the same input value (a landslide) in round 1 is:

\[
\text{Pr}[\text{landslide in round 1}] = \frac{1}{2^n}
\]

> What can we say about the following rounds?

1: \(a_p\) := input bit; \(n := 1\)
2: repeat forever
3: {phase 1}
4: send \(a_p\) to all
5: Let \(A\) be the multiset of the first \(n - f\) values with timestamp \(r\) received
6: if \(\exists v \in \{0, 1\} : \forall a \in A : a = v\) then \(a_p := v\)
7: else \(a_p := 1\)
8: {phase 2}
9: send \(a_p\) to all
10: Let \(B\) be the multiset of the first \(n - f\) values with timestamp \(r\) received
11: if \(\exists v \in \{0, 1\} : \forall b \in B : b = v\) then \(b_p := v\)
12: else if \(\exists b \in B : b \neq \perp\) then \(b_p := 6\)
13: else \(b_p := \perp\) \(\{ b \) is chosen uniformly at random to be 0 or 1\}
14: \(r := r + 1\)

Termination II

> In round \(r > 1\), the \(a\)-values are not necessarily chosen at random!

> By line 12, some process may set its \(a\)-value to a non-random value \(v\)

> By the Lemma, however, all non-random values are identical!

> Therefore, in every \(r\) there is a positive probability (at least \(\frac{1}{2^n}\)) for a landslide

> Hence, for any round \(c\)

\[
\text{Pr}[\text{no landslide at round } r] \leq 1 - \frac{1}{2^n}
\]

Since coin flips are independent:

\[
\text{Pr}[\text{no landslide for first } k \text{ rounds}] \leq (1 - \frac{1}{2^n})^k
\]

When \(k = 2^n\) this value is about \(\frac{1}{e}\); then, if \(k = c2^n\)

\[
\text{Pr}[\text{landslide within } k \text{ rounds}] \geq 1 - (1 - \frac{1}{2^n})^k \approx 1 - \frac{1}{e^c}
\]

which converges quickly to 1 as \(c\) grows.