Principles of Distributed Computing

Wireless Protocols

Yvonne-Anne Pignolet, May 2013

Wireless Networks

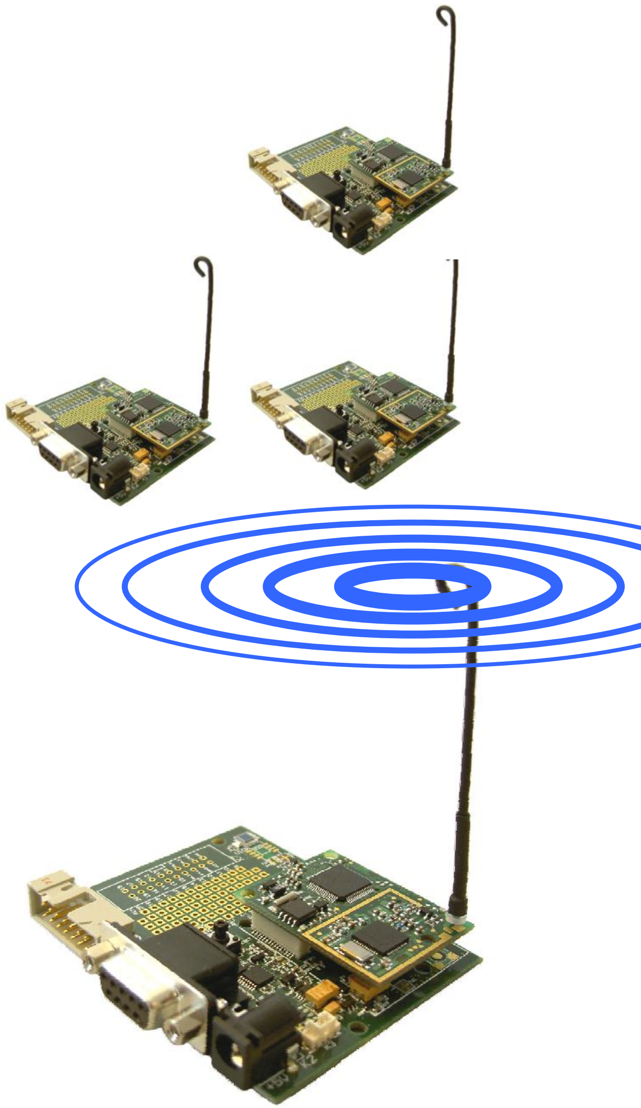
Very popular !

Biggest Advantage:

No wires ☺
=> fast installation
=> cheaper

Biggest Disdvantage:

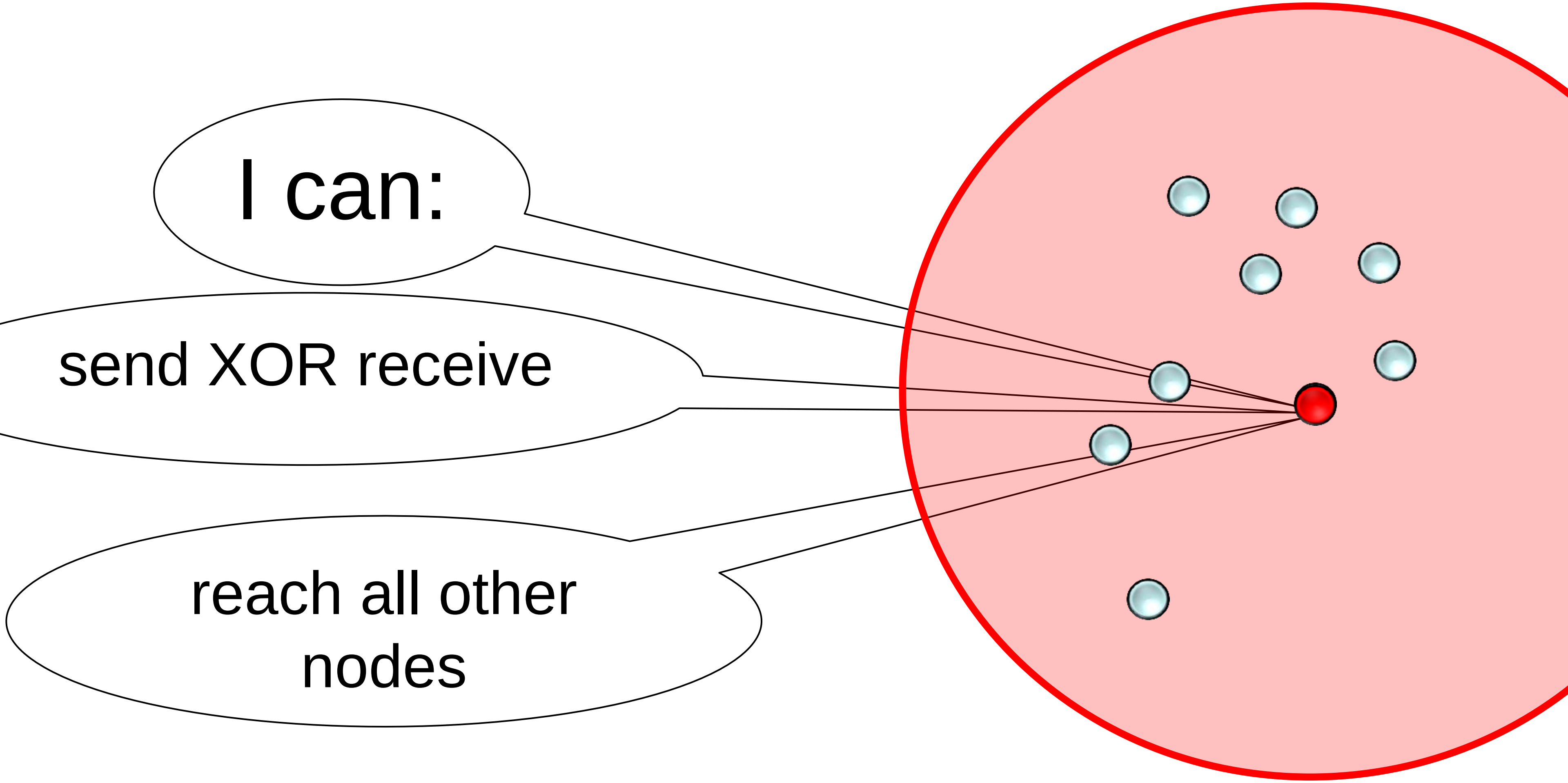
No wires ☺
=> attenuation
=> interference
=> energy supply



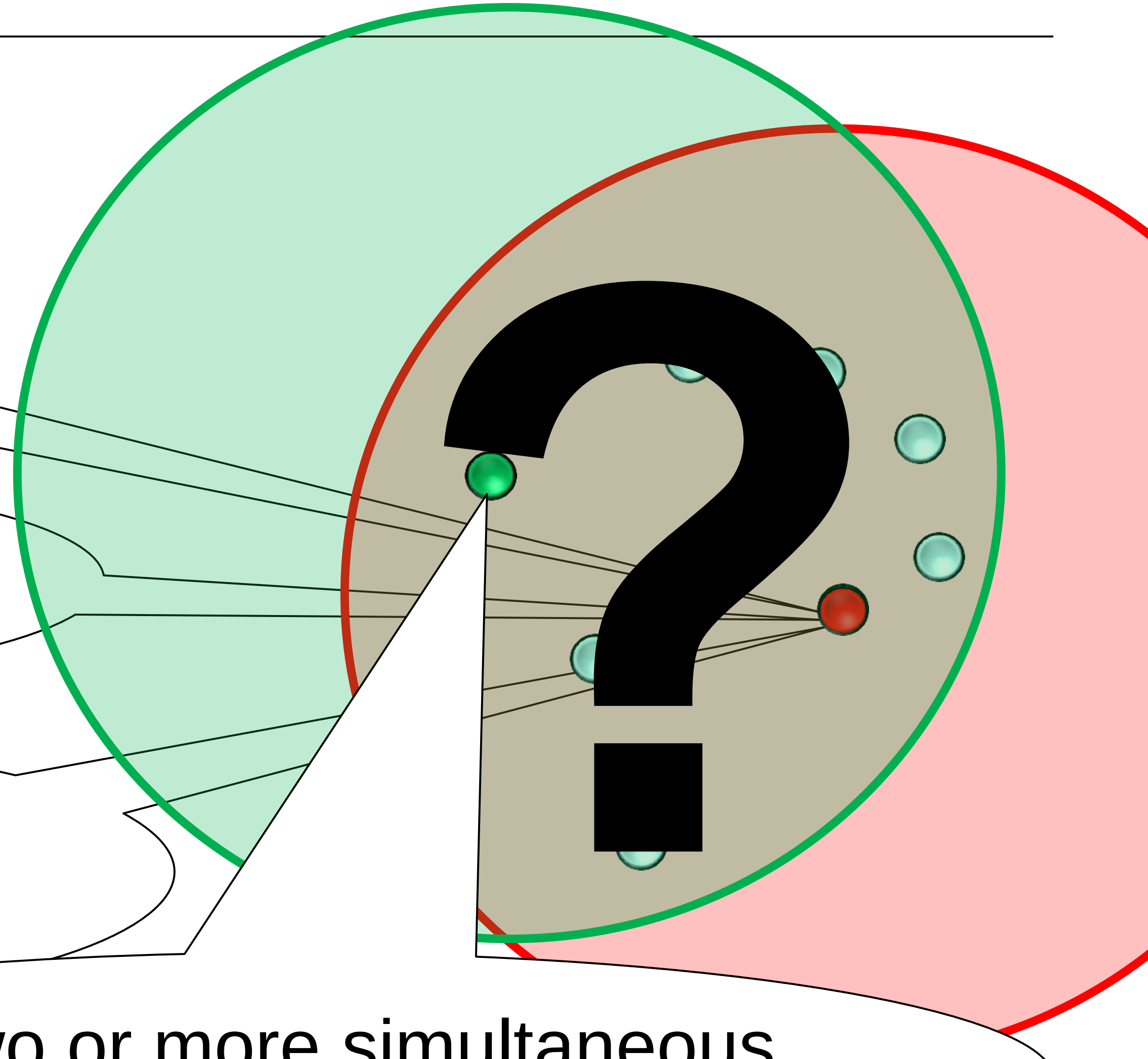
Big Question

To send or not to send?

Radio Network Model



Radio Network Model



I can:

send XOR receive

reach all other nodes

But two or more simultaneous transmissions collide

Today

Leader Election

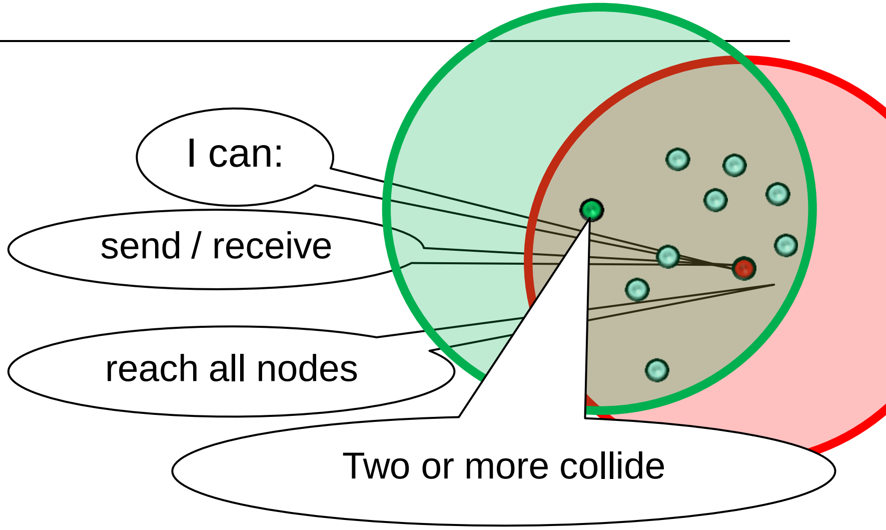
How long does it take until one node can transmit alone?

Initialization

How to assign IDs {1, 2, ….., n}?

Asynchronous Wakeup

How long for leader election if nodes wakeup up at arbitrary times?



With and without collision detection….

Def: X

X is the RV denoting the number of nodes transmitting in a given time slot

Leader Election without CD: Slotted Aloha

Slotted Aloha

repeat
    transmit with probability 1/n
until one node has transmitted alone

Expected time complexity: e

$$Pr[X=1] = n \cdot \frac{1}{n} \cdot \left(1 - \frac{1}{n}\right)^{n-1} \approx \frac{1}{e};$$

But then, how can the leader know its role?

The nodes start sending the ID of the leader with 1/n

But how can the node that sent the leader ID know the leader knows?

The leader sends an acknowledgement to this node.

Distributed ACK

Leader Election without CD: Unslotted Aloha

Slotted Aloha

repeat
    transmit with probability 1/n
until one node has transmitted alone

And without time slots?

$\Rightarrow$ Two partially overlapping messages collide
$\Rightarrow$ Probability for success drops to 1/(2e)

Why? Each slot is divided into t small time slots, $t \rightarrow \infty$,
nodes start a new t-slot long transmision with probability 1/(2nt)

Non-Uniform Initialization without CD

Repeated Aloha

i = 1
repeat
    transmit with probability 1/n
    if node v transmitted alone, v gets ID i, i++, n--
until all nodes have an ID

Each ID assignment takes expected time e
⇒ Total expected time n*e = O(n)

But:
Nodes need to known n!!!

Uniform Initialization with CD

Uniform Initialization

Subroutine Split(l)
repeat
    choose r uniformly at random from {0, 1}, join P_{l+r}
    in the next two time slots transmit in slot r and listen in other slot
until there was at least one transmission in both slots

Initialize()
N:= 1; L := 1;
while L $\geq$ 1 **do**
    all nodes in P_L transmit
    if exactly one node v has transmitted **then**
        v gets ID N and stops the protocol
        N++; L--;
    else
        use Split(L) to partition P_L into non-empty sets P_L and P_{L+1}
        L++
end while

Uniform Initialization with CD

Successful:
split into 2 non-empty subsets

Uniform Initialization

Subroutine Split(l)
repeat
    choose r uniformly at random from {0, 1}, join P_{l+r}
    in the next two time slots transmit in slot r and listen in other slot
until there was at least one transmission in both slots

Initialize()
N:= 1; L := 1;
while L $\geq$ 1 **do**
  all nodes in P_L transmit
  if exactly one node v has transmitted **then**
    v gets ID N and stops the protocol, N++; L--;
  else
    use Split(L) to partition P_L into non-empty sets P_L and P_{L+1}
    L++
end while

We need 2n-1 successfull splits ≈ creating a binary tree with n leaves and n-1 inner nodes.

Probability to create two non-empty subsets from a set of size k:

$$Pr[1 \leq X \leq k-1] = 1 - Pr[X=0] - Pr[X=k] = 1 - \frac{1}{2^k} - \frac{1}{2^k} \geq \frac{1}{2}.$$

Thus we need time O(n) for 2n-1 splits in expectation.

(with Chernoff whp)

Uniform Initialization without CD

Uniform Initialization (no CD)

1. Elect a leader
2. Divide every slot of the protocol with CD into two slots
   a) In the first slot, the nodes S transmit according to the protocol
   b) In the second slot, the nodes S from a) and the leader transmit
3. Distinguish the cases according to the table

   noise / silence : ✗

   successful transmission: ✔

	nodes in S transmit	nodes in $S \cup \{\ell\}$ transmit
$\|S\| = 0$	✗	✔
$\|S\| = 1, S = \{\ell\}$	✔	✔
$\|S\| = 1, S \neq \{\ell\}$	✔	✗
$\|S\| \geq 2$	✗	✗

Overhead: factor 2

More generally, a leader brings CD to any protocol

Leader Election With High Probability

Def: whp

An event happens with high probability if it occurs with
$p \geq 1\text{-}1/n\text{^}c$ for some constant c.

Slotted Aloha

repeat
    transmit with probability 1/n
until one node has transmitted alone

The probability of not electing a leader after c*log n time slots of Slotted Aloha is

$$\left(1-\frac{1}{e}\right)^{c\ln n} = \left(1-\frac{1}{e}\right)^{e\cdot c'\ln n} \leq \frac{1}{e^{\ln n\cdot c'}} = \frac{1}{n^{c'}}.$$

Uniform Leader Election (no CD)

Decrease Prob

for $k = 1, 2, 3, \ldots$ **do**
  for $i = 1$ **to** ck **do**
    transmit with probability $p := 1/2^k$
    if node v was the only node which transmitted **then**
      v becomes the leader
      break
    end if
  end for
end for

At the beginning: p too high and many collisions

When k ≈ log n, then p ≈ 1/n …
and we have a leader whp when i = O(log n) (see previous slide)

⇒ Time complexity O(log n * log n) = O(log^2 n)

Uniform Leader Election (with CD)

Transmit or keep silent

repeat
  transmit with probability $\frac{1}{2}$
  if at least one node transmitted **then**
    all nodes that did not transmit quit the protocol
  end if
until one node transmits alone

~ half of the nodes will never transmit again

active nodes decreases monotonically, but always ≥ 1.

Successful round (SR): at most half of active nodes transmit

Assume k ≥ 2 (otherwise we have elected a leader), then prob of SR:

$$Pr[1 \leq X \leq \lceil \frac{k}{2} \rceil] \geq \frac{1}{2} - Pr[X = 0] = \frac{1}{2} - \frac{1}{2^k} \geq \frac{1}{4}.$$

O(log n) SR for leader election. With Chernoff we can prove whp.

Faster Uniform Leader Election (with CD)

Guess, guess, walk

1. Get raw estimate of n, i ≈ (1 ± ½) log n
2. Get better estimate with binary search, j ≈ log n ± log log n
3. Do a biased random walk, k ≈ log n ± 2

1.

```
i := 1
repeat
   i := 2 · i
   transmit with probability 1/2^i
until no node transmitted
```

2.

```
u := 2^i      l := 2^(i-2)
while l + 1 < u do
   j := ⌈(l+u)/2⌉
   transmit with probability 1/2^j
   if no node transmitted then
      u := j
   else
      l := j
```

3.

```
k := u
repeat
   transmit with probability 1/2^k
   if no node transmitted then
      k := k - 1
   else
      k := k + 1
   end if
until exactly one node transmitted
```

$$\textit{If } j > \log n + \log\log n, \textit{ then } Pr[X > 1] \le \frac{1}{\log n}.$$

$$\textit{If } j < \log n - \log\log n, \textit{ then } P[X = 0] \le \frac{1}{n}.$$

$$\textit{If } i > 2\log n, \textit{ then } Pr[X > 1] \le \frac{1}{\log n}.$$

$$\textit{If } i < \frac{1}{2}\log n, \textit{ then } P[X = 0] \le \frac{1}{n}.$$

⇒ Time for Phase 1: O(log log n) with probability > 1-1/log n
⇒ Time for Phase 2: O(log log n) with probability > 1-1/log n

Faster Uniform Leader Election (with CD)

Guess, guess, walk

1.

$i := 1$
repeat
  $i := 2 \cdot i$
  transmit with probability $1/2^i$
until no node transmitted

2.

$u := 2^i$ $l := 2^{i-2}$
while $l + 1 < u$ **do**
  $j := \lceil \frac{l+u}{2} \rceil$
  transmit with probability $1/2^j$
  if no node transmitted **then**
    $u := j$
  else
    $l := j$

3.

$k := u$
repeat
  transmit with probability $1/2^k$
  if no node transmitted **then**
    $k := k - 1$
  else
    $k := k + 1$
  end if
until exactly one node transmitted

$i \approx (1 \pm \frac{1}{2}) \log n$ $j \approx \log n \pm \log \log n$ $k \approx \log n \pm 2$

Let v *be such that* $2^{v-1} < n \leq 2^v$, *i.e.,* $v \approx \log n$. *If* $k > v + 2$, *then* $Pr[X > 1] < \frac{1}{4}$.

If $k < v - 2$, *then* $P[X = 0] \leq \frac{1}{4}$.

If $v - 2 \leq k \leq v + 2$, *then* P[X = 1] is constant

⇒ Time for Phase 3: O(log log n) with probability > 1-1/log n (Chernoff)

⇒ Total time: O(log log n) with probability > 1- log log n/log n (union bound to keep error probability low)

Leader Election Lower Bound

Any uniform protocol with election probability of at least 1-1/2^t must run for at least t time slots.

For 2 nodes, the probability that exactly one transmits is at most $P[X = 1] = 2\,p\,(1 - p) \leq 1/2$.

Thus after time t the election probability is at most 1-1/2^t.

If a network with more than 2 nodes could find a leader quicker of with higher probability then so could 2 nodes.

Wakeup Lower Bound

Any uniform protocol has time complexity Ω(n/log n) for leader election whp if nodes wake up arbitrarily.

Whp unsuccessful

Uniform => all nodes executed the same code
At some point the nodes must transmit.

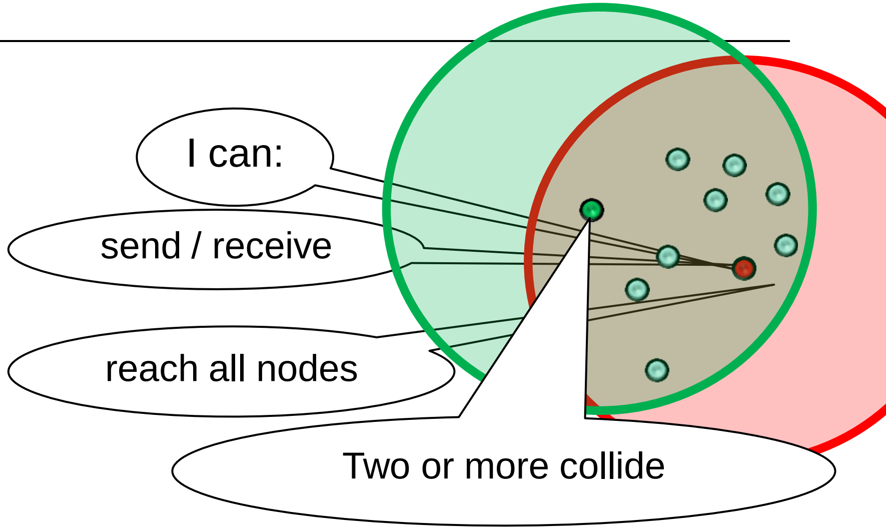
First transmission at time t, with probability p, independent of n
Adversary wakes up $w = \frac{c}{p} \ln n$ nodes in each time slot

Pr[E1] = P[X=1 at time t] $< \frac{1}{n^{c-1}} = \frac{1}{n^{c'}}$.

P[X !=1 at time t and the following n/w time slots]

$$= (1 - Pr(E_1))^{n/w} > \left(1 - \frac{1}{n^{c'}}\right)^{\Theta(n/\log n)} > 1 - \frac{1}{n^{c''}}.$$

Summary



Leader Election

How long does it take until one node can transmit alone?

- e in expectation, knowing n
- O(log n) whp, without knowing n, no CD
- O(log log n) without knowing n, with CD,
  with probability 1-loglog n/log n
- 1-1/log n election probability lower bound for O(log log n) time

Initialization

How to assign IDs {1, 2, ….., n}?

- O(n) with SplitInitialize (whp with Chernoff)

Asynchronous Wakeup

How long for leader election if nodes wakeup up at arbitrary times?

- Ω(n/log n) without IDs and without knowing n