Nodes in a distributed network may be anonymous (e.g., Lego MindStorm chips), or transmitting identities may be too expensive (e.g., FireWire bus).

Assumptions:

- Communication between nodes is asynchronous.
- Nodes have no identities, and carry the same local algorithm.

When a leader is known, all nodes can be named (using for instance a depth-first traversal).
**Theorem:** There is no terminating algorithm for electing a leader in an anonymous asynchronous network.

**Proof:** Take a (directed) ring of size \( N \).

In a symmetric configuration, all nodes are in the same state and all channels carry the same messages.

- The initial configuration is symmetric.

- If \( \gamma_0 \) is symmetric and \( \gamma_0 \rightarrow \gamma_1 \), then \( \gamma_1 \rightarrow \gamma_2 \rightarrow \cdots \rightarrow \gamma_N \) where \( \gamma_N \) is symmetric.
Probabilistic Algorithms

In a probabilistic algorithm, the execution of a node can be influenced by flipping a coin.

A probabilistic algorithm is Las Vegas if:

- the probability that it terminates is greater than zero; and
- all terminal configurations are correct.

Even if the probability that the algorithm terminates is 1, this does not imply termination.
Chang-Roberts Algorithm

Consider a directed ring of size $N$, with a total ordering on identities of nodes. Each node has a unique identity.

Each node sends out its identity; the smallest identity is the only one making a round trip.

- Each node $u$ sends its identity to its next neighbour.
- When $u$ receives $v$ with $v < u$, $u$ becomes passive and passes on the message $v$.
- When $u$ receives $v$ with $v > u$, $u$ purges the message.
- When $u$ receives $u$, it becomes the leader.

Worst-case message complexity: $O(N^2)$
Average-case message complexity: $O(N \log N)$
Example:

- Clockwise: \( \frac{1}{2} N(N + 1) \) messages
- Anti-clockwise: \( 2N - 1 \) messages
Itai-Rodeh Election Algorithm

Consider an anonymous, directed ring.
*Let all nodes know the ring size* $N$.

Each node selects a random identity from $\{1, \ldots, N\}$. Now run the Chang-Roberts algorithm.

**Complication:** Different nodes may select the same identity.

**Solution:** Each message is supplied with a hop count. A message arrives at its source if and only if its hop count is $N$.

When a node received a message with its own identity but a hop count $< N$, it passes on the message with a dirty bit.
If several nodes selected the same smallest identity, they start a fresh election round, **at a higher level**.

The Itai-Rodeh election algorithm is a **Las Vegas** algorithm; it **elects one leader with probability 1**.

**Average-case message complexity:** $O(N \log N)$

Without levels, the algorithm would break down.

**Example:**

\[
\begin{align*}
&u \quad u \\
&v < w, x \\
&u < v
\end{align*}
\]

\[
\begin{align*}
&w \\
&v < w, x
\end{align*}
\]
When channels are **FIFO**, round numbers are not needed.

Then the Itai-Rodeh algorithm becomes finite-state.

We made two versions, one with dirty bits (**Algorithm A**), and one without (**Algorithm B**).

We specified these algorithms as a **Markov decision process**, and performed a **model checking** analysis using **PRISM**.

They are Las Vegas algorithms that elect one leader with probability 1.
3 processes, 3 identities

probability of electing a leader vs. number of discrete time steps

- Algorithm A
- Algorithm B
Franklin’s Algorithm

Consider an undirected ring. Each node has a unique identity.

Each active node compares its identity with the identities of its nearest active neighbors. If its identity is not the smallest, it becomes passive.

- Each active node sends its identity to its neighbors.
- Let active node $u$ receive $v$ and $w$:
  - if $\min\{v, w\} < u$, then $u$ becomes passive
  - if $\min\{v, w\} > u$, then $u$ sends its identity to its neighbors again
  - if $\min\{v, w\} = u$, then $u$ becomes the leader
- Passive nodes pass on incoming messages.

Worst-case message complexity: $O(N \log N)$
Example:

![Diagram](image)

- Node labels: 0, 1, 2, 3, 4, 5
- Node connections
- Node 0 is labeled as the "leader"
Probabilistic Franklin Algorithm

Rena Bakhshi, Wan Fokkink and Jun Pang
Leader Election in Anonymous Rings: Franklin Goes Probabilistic
Under submission

In the probabilistic Franklin algorithm, again identities are selected at random, and a hop count is used to detect identity clashes.

Round numbers modulo 2 suffice! (With non-FIFO channels.)

With a $\mu$CRL model checking analysis we found that round numbers cannot be omitted altogether.
Invariants

For the probabilistic Franklin algorithm, the following invariant holds:

Between each pair of active nodes $u$, $v$ there are exactly two messages $m_1$, $m_2$.

If $m_1$, $m_2$ travel in opposite directions, then $u$, $v$, $m_1$, $m_2$ all carry the same bit as round number.

If $m_1$, $m_2$ travel in the same direction, then $u$, $v$ have opposite bits, and $m_1$, $m_2$ have opposite bits.

With a $\mu$CRL model checking analysis, up to ring size six, we verified that the probabilistic Franklin algorithm is a Las Vegas algorithm; it elects one leader with probability 1.

We used a distributed cluster of computers for this analysis.
Probabilistic Dolev-Klawe-Rodeh Algorithm

Dolev, Klawe and Rodeh (and independently Peterson) showed that Franklin’s idea can be implemented in a directed ring.

Then the comparison of identities of an active node $u$ and its nearest active neighbors $v$ and $w$ is performed at $w$.

\[
\begin{array}{c}
- & \rightarrow & v & - & \rightarrow & u & - & \rightarrow & w & - & \rightarrow
\end{array}
\]

In the probabilistic Dolev-Klawe-Rodeh algorithm, again identities are selected at random, and hop counts are used.

With a $\mu$CRL model checking analysis (on the distributed cluster) we found that in this case round numbers modulo 2 do not suffice.
Ring Size

There is no Las Vegas algorithm to compute the size of an anonymous ring!

Itai and Rodeh gave a Monte Carlo algorithm to compute the size of an anonymous ring. Such an algorithm may terminate incorrectly.

Again, each node selects identities at random, and a hop count is used to detect identity clashes.

The chance of terminating correctly can become arbitrarily close to 1 (by choosing node identities from a larger and larger domain).

When a leader has been elected, the ring size can be computed in a straightforward fashion.
**Leader Oracle**

Philippe Duchon, Nicolas Hanusse and Sébastien Tixeuil, *Optimal Randomized Self-stabilizing Mutual Exclusion on Synchronous Rings*, DISC’2004

**Tokens** move left or right with probability 0.5, and *merge* when they meet. Eventually, one token remains.

**Problem**: Processes cannot detect whether one token remains.


Using a *leader oracle*, which eventually returns a unique leader, they give a terminating leader election algorithm (assuming fairness).

They leave as an open question whether this is possible without oracle.

**Answer**: Without ring size knowledge, no.

   With ring size knowledge, yes.
FireWire

IEEE Standard 1394, called FireWire, is a serial multimedia bus. It connects digital devices, which can be added and removed dynamically.

It includes a leader election algorithm for undirected, acyclic networks. (Cyclic networks result in a time-out.)

The network size is unknown to the nodes. Identities are not used.

When a node has one possible father, it sends a parent request to this neighbor. If the request is accepted, an acknowledgement is sent back.
**Root contention:** The last two fatherless nodes can send parent requests to each other simultaneously.

They *randomly* decide to immediately send a parent request again, or to wait some time for a parent request from the other node.

**Question:** Is it optimal to give a 50% chance that a short resp. long delay is chosen?

The leader election algorithm for FireWire is a *Las Vegas* algorithm; it *elects one leader with probability 1* (in the absence of cycles).

Mariëlle Stoelinga

Fun with FireWire: A Comparative Study of Formal Verification Methods Applied to the IEEE 1394 Root Contention Protocol

*Formal Aspects of Computing, 14(3):328–337, April 2003*